



Universidad
de O'Higgins
Escuela de Ingeniería
Ingeniería Civil Eléctrica

Análisis comparativo de algoritmos de interfaz simulados en sistemas eléctricos de potencia aplicados a una red federada de laboratorios de Hardware in the Loop.

Marcelo Meza Martínez
Profesor(a) guía: Alfonso Ehijo

Memoria para optar al título y/o grado de Ingeniero Civil Eléctrico

Rancagua, Chile
Junio, 2024

Dedicatoria

Esta memoria está dedicada a mi madre y a mi padre, que siempre estuvieron conmigo apoyándome hasta el final.

Agradecimientos

Para empezar, me gustaría agradecer a todos los que estuvieron conmigo durante estos años en la carrera. Agradezco a mis familiares, mi padre que me inspiró en estudiar esta carrera, mi madre que alentaba todos los días en seguir mi camino apoyándome hasta el final, A mis hermanos, a mis amigos que formé durante la universidad. Agradecer a mi profesor guía Alfonso Ehijo, que me ayudó en mis momentos más difíciles al momento de hacer la memoria y a mi compañero de carrera Jairo Millapán, que estuvo trabajando codo a codo en el proyecto y otras cosas más.

También agradecer a mis grupos de amigos con los que me junto que me ayudaron a distraerme y reír de vez en cuando, a mis amigos virtuales que hice durante el camino, que me acompañaron el año pasado.

Y por último, agradecer a Scope Lab, por dejarme ser parte del proyecto, que siga creciendo el conocimiento y el espíritu por lograr muchas cosas que fomentan el desarrollo ingenieril para el país.

Índice

Resumen	6
Introducción	7
Motivación	8
Limitaciones/alcances	9
Objetivo general	10
Objetivos específicos	10
Marco teórico y revisión de literatura	11
Sistemas Eléctricos de Potencia (SEP)	11
<i>Generación de Electricidad</i>	11
<i>Transmisión de Energía</i>	11
<i>Subestaciones Eléctricas</i>	12
<i>Distribución de Energía</i>	12
<i>Control y Monitoreo</i>	12
<i>Operación Segura y Confiable</i>	12
<i>Integración de Energías Renovables</i>	12
<i>GDRTCS (Geographically Distributed Real-Time Co-Simulation)</i>	12
Tipos de simulaciones en tiempo real	13
<i>SIL (Software in the loop)</i>	13
HIL (Hardware in the loop)	14
<i>PIL (Processor in the loop)</i>	14
<i>RTNS (Real-Time Network Simulation)</i>	14
<i>Co-simulación</i>	15
<i>PHIL (Power Hardware in the loop)</i>	15
Protocolos de comunicación en SEP	16
<i>IEC 61850</i>	17
<i>GOOSE (Generic Object-Oriented Substation Event)</i>	17
<i>Modbus</i>	18
<i>DNP3 (Distributed Network Protocol 3)</i>	18
<i>IEC 60870-5</i>	19
Algoritmos de interfaz	19
Interfaz en PHIL y GDRTCS	22
<i>A. Técnica del Modelo de Transformador Ideal (ITM)</i>	22
Simulaciones en Tiempo Real Distribuidas Geográficamente (GDRTCS)	23
Relación entre ITM y GDRTCS	23
<i>B. Técnica del Modelo de Transformador Ideal Avanzado (AITM)</i>	24
<i>C. Aproximación de Primer Orden Variante en el Tiempo (TFA)</i>	25
<i>D. Modelo de Línea de Transmisión (TLM)</i>	25
<i>E. Duplicación Parcial de Circuito (PCD)</i>	26
<i>F. Método de Impedancia de Amortiguación (DIM)</i>	27

<i>G. Técnica del Filtro de Corriente de Retroalimentación (FCF)</i>	27
Desafíos y Mejora de la Estabilidad	28
Capas Principales	29
Marco Metodológico	30
Simulación de algoritmos de interfaz en Simulink	30
VSS (Voltage Source Subsystem)	30
PPS (Power Processing Subsystem)	30
<i>Técnica del Modelo de Transformador Ideal (ITM)</i>	31
Partes del Modelo con ITM Implementado	32
Fuentes Controladas de Voltaje y Corriente:	32
Función de Transferencia:	32
Control de Retardos:	32
Aplicación en Simulaciones HIL y PHIL:	32
Interconexión de Simuladores Distribuidos	33
<i>Técnica del Modelo de Transformador Ideal Avanzado (AITM)</i>	37
Modelo de Línea de Transmisión (TLM)	38
<i>Duplicación Parcial de Circuito (PCD)</i>	38
<i>Método de Impedancia de Amortiguación (DIM)</i>	39
<i>Técnica del Filtro de Corriente de Retroalimentación (FCF)</i>	40
Red Federada de Simuladores Distribuidos Geográficamente	40
Construcción de una Red Federada de Simuladores	40
Co-Simulación Distribuida Geográficamente	42
Resultados de los algoritmos implementados	43
Resultados para el algoritmo ITM	43
Análisis de Resultados ITM	44
Resultados para el algoritmo AITM	44
Análisis de Resultados AITM	44
Resultados para el Algoritmo TLM	45
Análisis de Resultados TLM	45
Resultados para el Algoritmo PCD	46
Análisis de Resultados PCD	46
Resultados para el Algoritmo DIM	47
Análisis de Resultados	48
Resultados para el Algoritmo FCF	48
Análisis de Resultados FCF	49
Discusión y Análisis	50
1. Análisis Comparativo de los Algoritmos	50
ITM (Ideal Transformer Model):	50
AITM (Advanced Ideal Transformer Model):	51
TLM (Transmission Line Model):	51
PCD (Partial Circuit Duplication):	51
DIM (Damping Impedance Method):	51
FCF (Feedback Current Filter):	52

2. Implicaciones para la Co-Simulación Distribuida	52
Conclusiones	53
Referencias	55
Anexos	57
Tabla 3 Comparación del algoritmo de AITM con diferentes configuraciones.	71
Tabla 4 Comparación del algoritmo de TLM con diferentes configuraciones.	71
Tabla 5 Comparación del algoritmo de PCD con diferentes configuraciones.	72
Tabla 6 Comparación del algoritmo de DIM con diferentes configuraciones.	72
Tabla 7 Comparación del algoritmo de FCF con diferentes configuraciones.	73
Carta Gantt	73

Resumen

En esta memoria se realizó un análisis exhaustivo de diversos algoritmos de interfaz aplicados a sistemas eléctricos de potencia (SEP), con un enfoque particular en su implementación dentro de una red federada de laboratorios de Hardware-in-the-Loop (HIL). La co-simulación distribuida geográficamente es una técnica avanzada que permite interconectar simuladores ubicados en diferentes lugares para modelar y analizar de manera colaborativa sistemas eléctricos complejos, reproduciendo con precisión las interacciones entre diferentes componentes del sistema.

El estudio se centró en la comparación de seis algoritmos de interfaz: Ideal Transformer Model (ITM), Advanced Ideal Transformer Model (AITM), Transmission Line Model (TLM), Partial Circuit Duplication (PCD), Damping Impedance Method (DIM) y Feedback Current Filter (FCF). Cada uno de estos algoritmos fue implementado y evaluado en cuatro configuraciones distintas para analizar su rendimiento en términos de estabilidad, precisión y eficiencia computacional, considerando el impacto del retardo y la latencia en la comunicación entre simuladores distribuidos.

Las simulaciones se llevaron a cabo en Simulink, utilizando sistemas de simulación en tiempo real como OPAL-RT, lo que permitió replicar condiciones cercanas a la realidad. Los resultados obtenidos revelaron las ventajas y limitaciones de cada algoritmo en diferentes escenarios, destacando la importancia de seleccionar y configurar adecuadamente los algoritmos de interfaz para garantizar la coherencia y la estabilidad de las simulaciones en un entorno distribuido.

Este trabajo no sólo aporta conocimientos valiosos sobre el comportamiento de los algoritmos de interfaz en sistemas distribuidos, sino que también proporciona una base sólida para el desarrollo de futuras estrategias que optimicen la co-simulación en redes federadas de laboratorios HIL. Los hallazgos obtenidos son cruciales para avanzar en la integración y el control eficiente de sistemas eléctricos de potencia a gran escala, especialmente en contextos donde la precisión y la estabilidad son críticas.

Palabras clave: Co-simulación, Sistemas eléctricos de potencia(SEP), Algoritmos de interfaz(IA), Red federada de laboratorios, Simulación en tiempo real, Eficiencia de transmisión de potencia.

Introducción

En la actualidad, el sector energético está experimentando una transformación significativa, impulsada por la creciente necesidad de integrar tecnologías avanzadas y mejorar la eficiencia operativa. Este fenómeno es parte de una tendencia más amplia hacia la industria 4.0, donde las tecnologías de la información y comunicación (TIC) juegan un papel crucial. En este contexto, surge la importancia de las redes inteligentes o Smart Grids, que representan una evolución de las infraestructuras eléctricas tradicionales hacia sistemas más interconectados, automatizados y eficientes.

Las Smart Grids se caracterizan por su capacidad para gestionar de manera inteligente el flujo de electricidad, integrando de forma efectiva las energías renovables y permitiendo una comunicación bidireccional entre los generadores y los consumidores de energía. Esta transformación no solo busca mejorar la estabilidad y eficiencia de los sistemas eléctricos de potencia (SEP), sino que también pretende optimizar la gestión de la demanda y reducir el impacto ambiental.

Para enfrentar estos desafíos, las tecnologías de comunicación juegan un papel central en la interconexión y coordinación de los componentes de los SEP. Protocolos como IEC 61850, Modbus, DNP3 y IEC 60870-5 son fundamentales para garantizar la interoperabilidad y la eficiencia en la transmisión de datos. Estos protocolos permiten una integración fluida de diversos dispositivos y sistemas, facilitando la supervisión, control y protección de la red eléctrica.

Dentro de este marco, la co-simulación emerge como una herramienta poderosa para analizar y optimizar el comportamiento de los SEP en un entorno controlado. La co-simulación permite la interacción de múltiples modelos de simulación, cada uno representando diferentes aspectos del sistema, como la electrónica de potencia, los sistemas de control y las redes de comunicación. Esto es particularmente relevante en el estudio de redes federadas, donde la coordinación entre distintas entidades es crucial para el funcionamiento eficiente y seguro del sistema. La propuesta de crear una Red Federada de Laboratorios con la técnica HIL implica establecer comunicación entre laboratorios utilizando una variedad de protocolos estándar, como OPC, Modbus TCP/IP, DNP3, IEC 61850, Ethernet/IP, entre otros. Esta comunicación permitirá la simulación integrada de una red de energía a gran escala, con pruebas exhaustivas que contribuyan a mejorar el diseño, la operación y la seguridad de la red en condiciones reales.

En la presente memoria se llevará a cabo un análisis comparativo de distintos algoritmos de interfaz (IA) aplicados a sistemas eléctricos de potencia, enfocados en su implementación en una red federada de laboratorios de Hardware-in-the-Loop (HIL). La co-simulación distribuida geográficamente es un enfoque que permite que distintos simuladores ubicados en diferentes lugares colaboren para representar y analizar el comportamiento de sistemas eléctricos complejos, replicando de manera precisa las interacciones que ocurren en la realidad.

El objetivo principal es evaluar cómo los diferentes algoritmos de interfaz, tales como el Ideal Transformer Model (ITM), el Advanced Ideal Transformer Model (AITM), el Transmission Line Model (TLM), el Partial Circuit Duplication (PCD), el Damping Impedance Method (DIM) y el Feedback Current Filter (FCF), influyen en la estabilidad, precisión y eficiencia de las simulaciones cuando se utilizan en una red de laboratorios distribuidos. Cada uno de estos algoritmos posee características particulares que los hacen más o menos adecuados para distintas situaciones de simulación, especialmente cuando se considera el desafío adicional de la latencia en la transmisión de datos entre los simuladores.

Motivación

El desarrollo de una Red Federada de Simuladores Distribuidos a lo largo del país, conectados y sincronizados a través de una variedad de protocolos de comunicación, representa una oportunidad significativa para avanzar en la gestión y optimización de los sistemas eléctricos de potencia. Esta red permitirá subdividir los sistemas de potencia, distribuyendo la carga computacional y permitiendo que múltiples simuladores, ubicados en diferentes puntos geográficos, trabajen de manera conjunta y coordinada. La motivación principal detrás de este estudio radica en la posibilidad de utilizar esta red para realizar simulaciones y pruebas a gran escala, replicando condiciones reales de operación y facilitando el desarrollo de soluciones más robustas y eficientes para los desafíos energéticos actuales.

Además, este enfoque podría reducir significativamente los costos asociados con la infraestructura de simulación, al permitir que se utilicen los recursos de cómputo de manera más eficiente y distribuida. La implementación de esta red federada no solo contribuiría al avance del conocimiento en el área de los sistemas eléctricos, sino que también tendría un impacto directo en la capacidad del país para integrar energías renovables, mejorar la estabilidad de la red y reducir las emisiones de carbono, todo ello mientras se fomenta la formación de ingenieros altamente capacitados en tecnologías de vanguardia.

Limitaciones/alcances

El trabajo se limita al uso de Simulink para la co-simulación, evaluando solo los algoritmos de interfaz, lo que puede restringir la generalización de los resultados. Sin embargo, establece una base sólida para futuras investigaciones y mejoras en la eficiencia operativa y estabilidad de los sistemas eléctricos.

Contribución

El estudio ofrece una evaluación detallada de algoritmos de interfaz en co-simulación, proporcionando una guía práctica para su uso. Contribuye al campo emergente de co-simulación en redes inteligentes y propone la creación de una Red Federada de Laboratorios usando HIL, facilitando la simulación y pruebas exhaustivas en redes eléctricas a gran escala, además de tener un impacto en la formación de ingenieros y el desarrollo de tecnologías sostenibles.

Objetivo general

- Analizar algoritmos de interfaz aplicados en co-simulaciones de sistemas eléctricos de potencias, implementando simulaciones de los distintos algoritmos aplicados para una red federada de laboratorios de simulación.

Objetivos específicos

- Investigar los protocolos de comunicación en sistemas electricos de Potencia (SEP)
- Simular algoritmos de interfaz dentro del contexto de la red federada de laboratorios
- Realizar un análisis comparativo de los algoritmos de interfaz y validar contra resultados

Marco teórico y revisión de literatura

Sistemas Eléctricos de Potencia (SEP)

Los Sistemas Eléctricos de Potencia (SEP) comprenden una red compleja de componentes, equipos y procesos destinados a la generación, transmisión y distribución de energía eléctrica. La energía disponible en la naturaleza se convierte, mediante diversas tecnologías, en energía eléctrica. Estas tecnologías incluyen fuentes hidráulicas, eólicas, térmicas, nucleares, solares y geotérmicas. Una vez generada, la electricidad es transportada, principalmente en forma de corriente alterna (CA), que se caracteriza por sus ondas senoidales. Estas ondas son esenciales no solo por su representación matemática sencilla, sino también porque reflejan muchos fenómenos naturales y también permiten un transporte de la energía más eficiente debido al uso de transformadores [1][10].

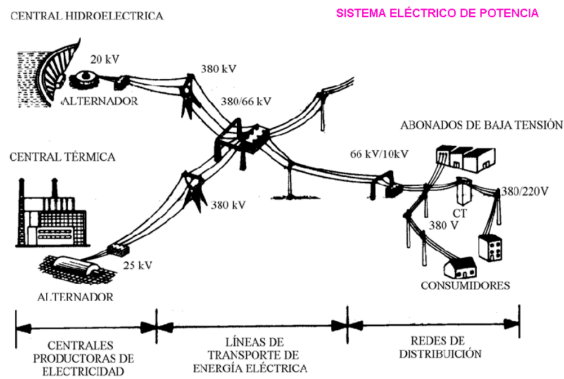


Fig 1. Arquitectura de un Sistema eléctrico de potencia, basado en [20].

Generación de Electricidad

Las centrales eléctricas constituyen la base de los SEP, produciendo electricidad a partir de diversas fuentes energéticas. Estas instalaciones generan energía en forma de CA, que es posteriormente distribuida a través de la red.

Transmisión de Energía

La electricidad generada es transmitida a largas distancias mediante líneas de alta tensión. Este transporte a elevados voltajes es crucial para minimizar las pérdidas de energía durante el trayecto, asegurando así la eficiencia en la entrega de electricidad.

Subestaciones Eléctricas

Durante el proceso de transmisión, la electricidad pasa por subestaciones, donde se realizan transformaciones de voltaje, control y protección. Estas estaciones son fundamentales para ajustar el nivel de tensión adecuado para su distribución a los usuarios finales.

Distribución de Energía

La energía eléctrica llega a los consumidores finales a través de redes de distribución. Estas redes incluyen líneas de distribución, transformadores y equipos de protección que garantizan una entrega segura y confiable de electricidad a hogares, industrias y comercios.

Control y Monitoreo

Los SEP cuentan con sofisticados sistemas de control y monitoreo que supervisan el flujo de energía, detectan fallas y aseguran la estabilidad del sistema. Estos sistemas emplean tecnologías avanzadas como la automatización, telemetría y comunicaciones en tiempo real para gestionar el sistema eficientemente.

Operación Segura y Confiable

Una de las funciones críticas de los SEP es mantener una operación segura y confiable. Esto incluye la coordinación precisa entre la generación y la demanda de energía, la protección contra sobrecargas y cortocircuitos, así como la gestión de situaciones de emergencia como apagones.

Integración de Energías Renovables

Con el crecimiento de las energías renovables, los SEP están adoptando tecnologías para integrar eficientemente fuentes como la solar y la eólica. Esto incluye el desarrollo de infraestructura para la conexión de parques eólicos y plantas solares, y la gestión de la intermitencia de estas fuentes.

GDRTCS (Geographically Distributed Real-Time Co-Simulation)

El GDRTCS (Geographically Distributed Real-Time Co-Simulation) es un enfoque avanzado en la simulación de sistemas eléctricos de potencia y otros sistemas complejos, que permite la integración y simulación coordinada de múltiples subsistemas distribuidos geográficamente. En este tipo de co-simulación, diferentes partes de un sistema o subsistemas se simulan en tiempo real utilizando simuladores ubicados en diferentes ubicaciones geográficas. Estos simuladores

están conectados a través de redes de comunicación, permitiendo la transmisión y sincronización de datos entre ellos.

La clave del GDRTCS radica en la capacidad de los sistemas de comunicarse y colaborar de manera eficiente y en tiempo real, a pesar de estar separados físicamente. Esto se logra utilizando protocolos de comunicación estandarizados y algoritmos de interfaz que aseguran la fidelidad de la simulación y minimizan los efectos de latencias y otros problemas de red. Los beneficios de GDRTCS incluyen la capacidad de simular escenarios complejos que involucran múltiples subsistemas, como redes eléctricas y sistemas de control distribuidos, y evaluar su comportamiento bajo condiciones operativas realistas y variadas.

Este enfoque es particularmente útil en estudios de integración de energías renovables, sistemas de almacenamiento de energía, y redes inteligentes, donde la interconexión de múltiples componentes y subsistemas es esencial para la operación eficiente y segura del sistema global. Además, facilita la colaboración entre diferentes instituciones y organizaciones, permitiendo que cada una utilice su infraestructura de simulación específica para contribuir a un estudio integral y exhaustivo.

Tipos de simulaciones en tiempo real

SIL (Software in the loop)

Es un tipo de simulación que permite probar códigos fuente, circuitos de control, sistemas eléctricos, y otros componentes de sistemas complejos dentro de un entorno computacional sin la necesidad de disponer de un equipo externo físicamente. Esta modalidad de simulación es especialmente útil en la etapa de desarrollo y validación de proyectos de ingeniería, ya que facilita la detección y corrección de errores antes de la implementación física. Además, permite realizar múltiples pruebas y ajustes de manera rápida y eficiente, optimizando el tiempo y los recursos empleados en el desarrollo.

Usando SIL, los ingenieros pueden realizar pruebas y ajustes directos de su código fuente en una PC, conectando el software a un modelo digital de la planta. Este modelo digital reemplaza a sistemas físicos, prototipos o bancos de prueba, que son más caros. Así, se puede probar el software antes de construir el hardware, acelerando el ciclo de desarrollo.

SIL ayuda a detectar defectos o errores en las etapas tempranas del desarrollo del sistema, lo que reduce los costos de resolver problemas en etapas posteriores, cuando las interacciones entre componentes son más complicadas. Es un complemento excelente para la

simulación Hardware-in-the-Loop (HIL) y contribuye a acelerar el lanzamiento al mercado y a desarrollar software de manera más eficiente.

HIL (Hardware in the loop)

El Hardware-in-the-Loop (HIL) es una técnica que utiliza piezas reales, como componentes eléctricos, en un entorno de simulación para probar cómo funcionan junto a otros sistemas en situaciones reales. Básicamente, se conecta un controlador a un sistema que simula el comportamiento de un producto completo, replicando condiciones del mundo real. Esto permite probar y ajustar el diseño como si los dispositivos estuvieran operando en la realidad, pero sin los costos y el tiempo que implican las pruebas físicas tradicionales. De este modo, se pueden ejecutar miles de escenarios diferentes de manera eficiente.

PIL (Processor in the loop)

Similar a Hardware in the Loop (HIL), Processor in the Loop (PIL) tiene tanto el sistema del controlador como la planta en el bucle de control, pero en este caso, ambos componentes son físicos. Esta metodología es especialmente adecuada para industrias de procesos, como las químicas o farmacéuticas, donde se prueban controladores en procesos industriales complejos y críticos. En estas industrias, los controladores deben manejar con precisión y fiabilidad diversas variables del proceso, como temperatura, presión y flujo, para asegurar la calidad y seguridad del producto final.

RTNS (Real-Time Network Simulation)

Es una simulación del comportamiento en redes de comunicaciones en tiempo real, diseñada para evaluar y analizar el rendimiento de las redes bajo diversas condiciones operativas. Este tipo de simulación es especialmente útil para probar el rendimiento de redes en situaciones de baja latencia y tráfico variable, permitiendo a los ingenieros y desarrolladores identificar y resolver potenciales problemas antes de su implementación en entornos reales.

En la simulación de redes en tiempo real, se pueden replicar múltiples escenarios, como congestión de tráfico, fallos de enlace, interferencias, y cambios dinámicos en la topología de la red. Estos simuladores pueden modelar tanto redes alámbricas como inalámbricas, y son cruciales para entender cómo diferentes protocolos y configuraciones de red afectarán el rendimiento global.

Co-simulación

Como sugiere el nombre, las co-simulaciones integran múltiples simuladores en una sola plataforma de simulación, permitiendo la interacción y colaboración entre diferentes modelos y herramientas de simulación. Esta técnica es especialmente valiosa en proyectos de ingeniería complejos que requieren la interacción precisa y coordinada entre múltiples subsistemas.

Las co-simulaciones facilitan la integración de diversos dominios de simulación, como la mecánica, la electrónica, la hidráulica, y la aerodinámica, en un entorno unificado. Esto permite a los ingenieros y desarrolladores analizar el comportamiento conjunto de estos subsistemas en tiempo real, identificando posibles problemas de interoperabilidad y optimizando el rendimiento global del sistema.

PHIL (Power Hardware in the loop)

La simulación Power Hardware-in-the-Loop (PHIL) es una evolución del Hardware-in-the-Loop (HIL), que permite el intercambio de potencia real en lugar de solo señales de baja tensión y corriente. En PHIL, amplificadores de potencia se utilizan para conectar dispositivos de prueba de alta potencia con el simulador, asegurando un correcto cierre del lazo mediante retroalimentación. Estos amplificadores se seleccionan en función de su rendimiento y capacidad para manejar energía [18].

PHIL permite la simulación de flujos de alta potencia, facilitando la prueba de sistemas como convertidores de potencia, generadores y motores, con mayor fidelidad, flexibilidad y seguridad en comparación con los métodos tradicionales.

Los modelos de alta fidelidad en tiempo real de sistemas de potencia y electrónica de potencia, cuando se combinan con un amplificador de calidad, pueden ofrecer un rendimiento similar al de un dinamómetro o un banco analógico. Un dinamómetro es un dispositivo clave para medir el torque y la potencia de un motor o sistema en funcionamiento, mientras que un banco de pruebas analógico es una instalación completa utilizada para evaluar el rendimiento de motores y sistemas bajo condiciones controladas. Ambos son fundamentales en la industria y la investigación para garantizar que los motores y sistemas funcionen de manera óptima y segura. Esto permite alcanzar un nivel de confianza del 90% en que el sistema funcionará como se espera [19].

- a. La simulación Power Hardware-in-the-Loop (PHIL) permite a los desarrolladores probar una mayor variedad de características en comparación con los bancos analógicos o los dinamómetros, y además requiere menos mantenimiento y

configuración. Esto hace que el hardware sea más robusto frente a una amplia gama de parámetros, características y fallos.

- b. Además de su capacidad para interactuar con el hardware, los simuladores PHIL se utilizan frecuentemente para simular redes de comunicación como CAN, DNP3, ARINC, 61850, entre otras. Esto permite la integración de múltiples protocolos y sistemas en un solo sistema.
- c. Un simulador PHIL proporciona un sistema robusto, flexible, versátil y confiable, lo que permite realizar múltiples experimentos para diversos programas o investigaciones.

Protocolos de comunicación en SEP

En el contexto de los Sistemas Eléctricos de Potencia (SEP), la comunicación entre diversos componentes es crucial para la operación eficiente y segura de la red eléctrica. Los protocolos de comunicación no solo facilitan la transferencia de datos y la coordinación de dispositivos, sino que también soporta funciones avanzadas como la protección, el control y la supervisión en tiempo real. Esto es especialmente relevante en el contexto de co-simulación, donde se integran múltiples modelos y sistemas para simular escenarios complejos y evaluar el desempeño del sistema en su totalidad. A continuación, se detallan los principales protocolos de comunicación utilizados en SEP, con un enfoque en sus características técnicas y aplicaciones [2][3][4].

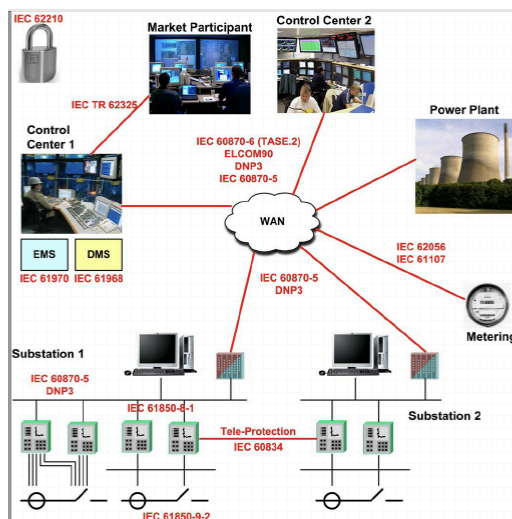


Fig 2. Diferentes Tipos de Protocolos de comunicación, basado en [21].

IEC 61850

El IEC 61850 es un estándar internacional desarrollado para las comunicaciones de redes y sistemas en subestaciones. Diseñado para unificar y estandarizar los diversos protocolos utilizados en los SEP, IEC 61850 se destaca por su enfoque en la interoperabilidad. Este protocolo permite la comunicación entre dispositivos de diferentes fabricantes, integrando protocolos como IEC 60870-5-101/104, Modbus y DNP3, además de soportar protocolos propietarios. Esto facilita no solo la interoperabilidad, sino también la sustitución y actualización de equipos sin restricciones de compatibilidad [5].

Una de las características distintivas de IEC 61850 es su modelo de datos basado en objetos y funciones, que simplifica la organización y gestión de la información dentro de las subestaciones. Este modelo permite la definición de nodos lógicos y unidades de información simples, proporcionando una estructura clara y fácil de manejar para los ingenieros. Además, la capacidad de auto-descripción de los dispositivos mejora significativamente los procesos de ingeniería y mantenimiento, permitiendo configuraciones y puestas en marcha más rápidas y menos propensas a errores [13].

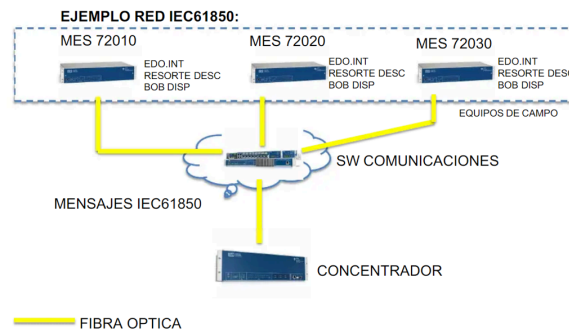


Fig 3. Ejemplo de red IEC 61850, basado en [11].

En el ámbito de la co-simulación, IEC 61850 es particularmente útil porque soporta la integración de múltiples sistemas y simuladores, facilitando la evaluación de escenarios complejos donde se deben coordinar varios subsistemas. La capacidad de definir archivos de configuración de subestaciones (SCD) utilizando un lenguaje basado en XML permite la simulación y el análisis de diferentes configuraciones de red de manera eficiente [11][14].

GOOSE (Generic Object-Oriented Substation Event)

Dentro del estándar IEC 61850, el protocolo GOOSE es esencial para la comunicación de eventos en tiempo real entre dispositivos electrónicos inteligentes (IEDs). Este protocolo utiliza un mecanismo de comunicación de tipo editor/suscriptor, lo que permite la transmisión rápida

de mensajes de control y protección en situaciones críticas. El diseño basado en eventos de GOOSE asegura que los mensajes se envíen sólo cuando ocurren cambios significativos, lo que minimiza la latencia y garantiza una respuesta rápida [12].

En el contexto de co-simulación, GOOSE es valioso porque permite probar y validar las respuestas del sistema ante diferentes eventos de protección y control. La capacidad de simular eventos y observar cómo se propagan a través de la red permite a los ingenieros ajustar y optimizar los sistemas de protección antes de su implementación en el campo.

Modbus

Modbus es uno de los protocolos más antiguos y ampliamente utilizados en la industria de la automatización. Desarrollado originalmente para facilitar la comunicación entre controladores lógicos programables (PLC), Modbus ha evolucionado para soportar tanto conexiones seriales (Modbus RTU) como redes Ethernet (Modbus TCP/IP). Este protocolo es conocido por su simplicidad y flexibilidad, lo que lo hace ideal para aplicaciones de monitoreo y control en sistemas SCADA [6].

En términos de co-simulación, Modbus puede integrarse fácilmente en modelos de simulación para representar la comunicación entre dispositivos de campo y sistemas de control centralizados. Esto permite una evaluación detallada de cómo los datos se transmiten y se utilizan para tomar decisiones operativas. La capacidad de Modbus para gestionar bloques de datos sin restricciones es particularmente útil en escenarios donde se necesita una transferencia rápida y eficiente de grandes volúmenes de información.

DNP3 (Distributed Network Protocol 3)

DNP3 es un protocolo robusto diseñado específicamente para la comunicación en sistemas de control de infraestructura crítica. Este protocolo es eficiente en el uso del ancho de banda y puede operar en ambientes donde las condiciones de comunicación son desfavorables. DNP3 soporta la autenticación y el cifrado de datos, lo que lo hace adecuado para aplicaciones donde la seguridad de la información es una preocupación.

Una característica clave de DNP3 es su orientación a eventos, lo que significa que transmite datos sólo cuando hay cambios significativos en el sistema. Esto es especialmente útil en co-simulación para analizar cómo los eventos y las alarmas se manejan en la red, permitiendo a los ingenieros ajustar los sistemas de respuesta y control en función de diferentes escenarios [7].

IEC 60870-5

El IEC 60870-5 es un protocolo ampliamente utilizado en Europa para la comunicación entre sistemas SCADA y dispositivos de campo como RTUs y relés de protección. Este protocolo sigue una arquitectura maestro/esclavo, lo que permite una estructura de comunicación jerárquica donde un sistema centralizado puede controlar y monitorear múltiples dispositivos distribuidos [8][9].

Este protocolo es especialmente útil en aplicaciones de co-simulación para modelar la comunicación en redes eléctricas de gran escala. Su capacidad para soportar la transmisión de datos en tiempo real y la gestión de alarmas lo convierte en una herramienta valiosa para evaluar el desempeño del sistema bajo diferentes condiciones operativas. Además, su flexibilidad y adaptabilidad a diferentes configuraciones de red permiten una integración fácil en diversos modelos de simulación.

Los protocolos de comunicación en los SEP son fundamentales para la operación y supervisión eficaces de la red eléctrica. Desde la interoperabilidad facilitada por IEC 61850 hasta la robustez de DNP3 en entornos críticos, cada protocolo ofrece ventajas específicas que lo hacen adecuado para diferentes aplicaciones. En el contexto de co-simulación, estos protocolos no solo permiten una evaluación precisa y detallada de los sistemas, sino que también facilitan la integración de nuevas tecnologías y estrategias operativas. La correcta implementación y simulación de estos protocolos es esencial para desarrollar soluciones innovadoras que optimicen la gestión de los sistemas eléctricos de potencia.

Algoritmos de interfaz

Un algoritmo de interfaz (IA) es una metodología que permite la comunicación y el intercambio de señales entre diferentes simuladores en tiempo real que modelan distintos subsistemas de un sistema más grande. Estos algoritmos son esenciales para mantener la precisión y estabilidad de la simulación, especialmente en entornos distribuidos geográficamente. Un buen IA asegura que los subsistemas interactúen de manera coherente, manejando los posibles retrasos, el ruido y la variabilidad en las condiciones de la red de comunicación. Además, debe garantizar la conservación de la energía en la interfaz y minimizar errores, permitiendo que las simulaciones sean lo más realistas y estables posible.

En la co-simulación y en aplicaciones como Power Hardware-in-the-Loop (PHIL), los algoritmos de interfaz (IA) son esenciales para garantizar la interacción precisa y estable entre los dominios físico y virtual. Al conectar un Sistema Simulado Virtual (VSS) con un sistema real,

los IA ayudan a minimizar las inexactitudes y perturbaciones que pueden surgir debido al acoplamiento de elementos como sondas de medición y amplificadores de potencia. Estas inexactitudes, si no se manejan adecuadamente, pueden afectar la estabilidad y la fidelidad de los experimentos.

Los sistemas PHIL utilizan amplificadores de potencia y sensores que son cruciales para las pruebas, pero que también pueden introducir dinámicas no deseadas en el sistema. Para abordar estos desafíos, se han desarrollado diversos algoritmos de interfaz. Entre los más destacados se encuentran el Modelo de Transformador Ideal (ITM), el Modelo de Transformador Ideal Avanzado (AITM), la Aproximación de Primer Orden Variante en el Tiempo (TFA), el Modo de Línea de Transmisión (TLM), la Duplicación Parcial de Circuito (PCD), el Método de Impedancia de Amortiguación (DIM) y el Filtro de Corriente de Retroalimentación Extensible (FCF).

Cada uno de estos algoritmos tiene características específicas que los hacen adecuados para diferentes escenarios. Un buen IA debe no solo mejorar la precisión y estabilidad de la simulación, sino también, asegurar una interacción coherente entre los subsistemas simulados. En el entorno de co-simulación, donde se integran modelos físicos y simulados, es fundamental que los IA gestionen correctamente los desafíos como el retraso, el ancho de banda y el ruido en la transmisión de señales.

Los algoritmos de interfaz se clasifican según el tipo de señal que transmiten, como potencia, corriente, voltaje, fuerza o torque. Además, utilizan diversos métodos de procesamiento de señales para mejorar la calidad de la simulación. Entre estos métodos se incluyen el filtrado de paso bajo, que ayuda a eliminar el ruido de alta frecuencia, la ganancia unitaria, que asegura la consistencia de la señal, y los compensadores de adelanto-atraso, que ajustan la fase de la señal para mejorar la estabilidad del sistema.

Por lo tanto, los algoritmos de interfaz son una pieza clave en la co-simulación de sistemas eléctricos de potencia. Su correcta implementación y ajuste permiten realizar experimentos realistas y estables, proporcionando una plataforma robusta para la evaluación y desarrollo de nuevas tecnologías y estrategias operativas en el campo de la ingeniería eléctrica. Estos algoritmos permiten una integración efectiva de los elementos físicos y virtuales, asegurando que los resultados de la simulación sean representativos de las condiciones reales del sistema [17].

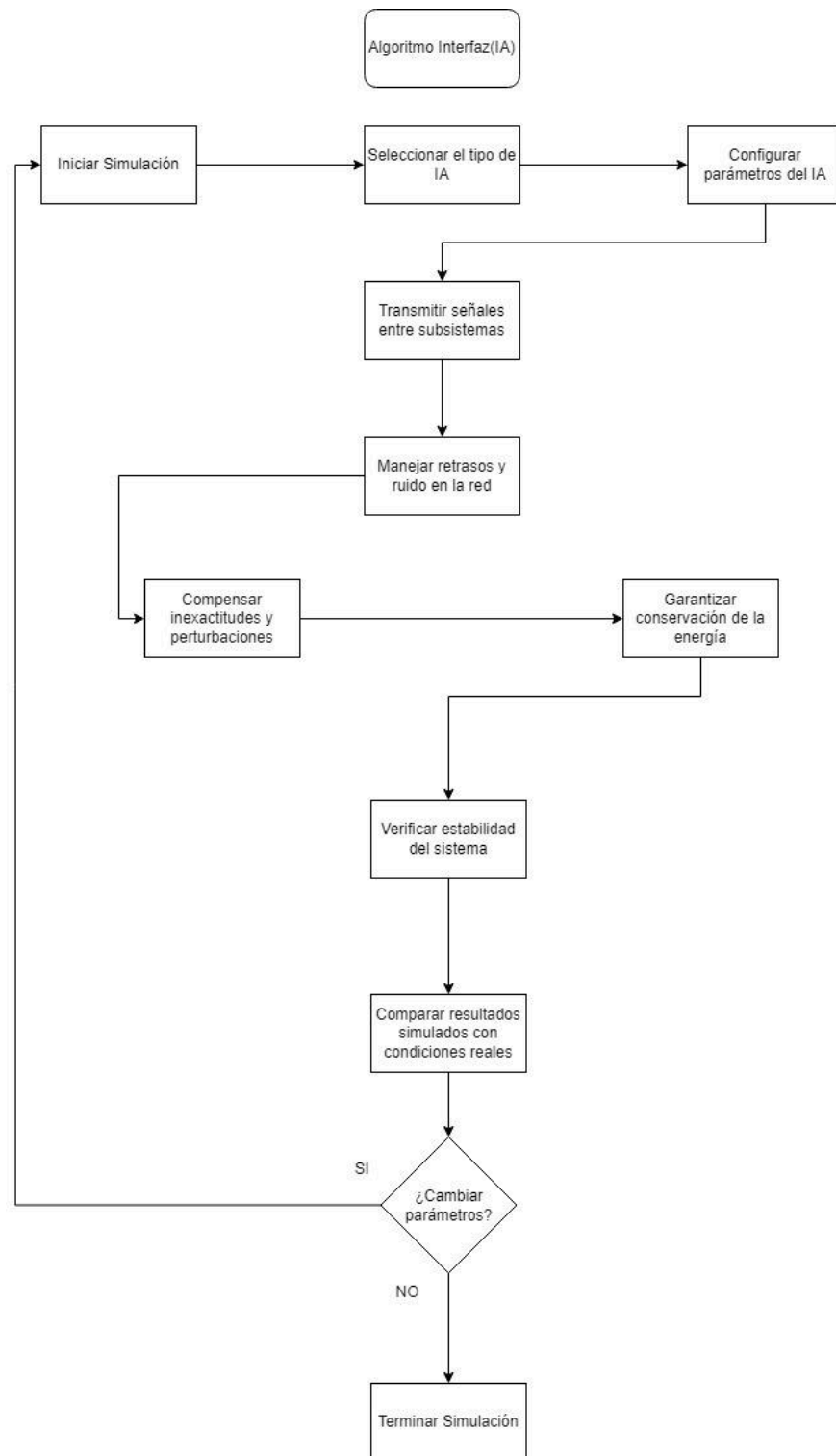


Fig 4. Diagrama de Flujo sobre un Algoritmo de Interfaz
Fuente: Elaboración propia

Interfaz en PHIL y GDRTCS

En un entorno PHIL, los subsistemas se conectan mediante un elemento de acoplamiento de potencia que facilita la interacción entre el sistema real y el sistema simulado virtual (VSS). Este acoplamiento es crucial para realizar experimentos que simulan de manera realista las condiciones del sistema físico, y su precisión depende en gran medida del diseño del algoritmo de interfaz. La arquitectura de un sistema PHIL se modela comúnmente como un equivalente de Thévenin, lo que requiere que el IA maneje con precisión las señales de potencia, corriente, voltaje y otras variables relevantes.

Los IA deben considerar varios factores, como el impacto de la red de comunicación, la conservación de la energía en la interfaz y la estabilidad global del sistema. La correcta implementación de un IA es fundamental para minimizar las inexactitudes y perturbaciones causadas por elementos como sondas de medición y amplificadores de potencia, los cuales pueden afectar la dinámica del sistema.

A. Técnica del Modelo de Transformador Ideal (ITM)

El Método del Transformador Ideal (ITM) es un enfoque crucial en la simulación distribuida de sistemas eléctricos, especialmente en el contexto de Simulaciones en Tiempo Real Distribuidas Geográficamente (GDRTCS, por sus siglas en inglés). Estas simulaciones permiten la evaluación y análisis de sistemas de potencia complejos, que pueden estar físicamente ubicados en diferentes lugares, pero que necesitan ser simulados de manera coordinada para entender su comportamiento global.

El ITM se basa en la analogía de un transformador ideal que permite la transferencia de voltaje y corriente entre subsistemas acoplados, como un generador y una carga, que pueden estar simulados en plataformas separadas o incluso en diferentes ubicaciones geográficas. Este método garantiza que la interacción entre estos subsistemas se realice de manera precisa, manteniendo la coherencia en las señales de voltaje y corriente mientras se transmite la energía eléctrica entre ellos.

El objetivo del ITM es dividir un sistema eléctrico en subsistemas manejables que pueden ser simulados por separado, pero que deben intercambiar información de manera precisa y sin pérdidas significativas. Esto es crucial en simulaciones distribuidas donde cada subsistema puede estar simulando una parte distinta del sistema eléctrico global.

Simulaciones en Tiempo Real Distribuidas Geográficamente (GDRTCS)

En el contexto de GDRTCS, los sistemas eléctricos son simulados en tiempo real en ubicaciones geográficas diferentes, permitiendo a múltiples centros de simulación trabajar en conjunto para modelar y analizar un sistema de potencia completo. Cada centro de simulación puede estar encargado de una parte específica del sistema, como la generación, transmisión o distribución de energía, y todos estos subsistemas deben interactuar de manera sincronizada.

La aplicación del ITM en GDRTCS es esencial para asegurar que la simulación distribuida sea coherente. El algoritmo permite que los voltajes y corrientes calculados en un subsistema sean transformados y enviados a otro subsistema, de tal manera que se mantenga la estabilidad y la precisión de la simulación global. Esto es especialmente importante cuando los subsistemas operan en diferentes entornos o plataformas de simulación que deben comunicarse a través de redes con posibles retardos y pérdidas de datos.

Relación entre ITM y GDRTCS

El ITM facilita la interfaz entre subsistemas en un entorno de simulación distribuida al modelar la interacción entre diferentes partes del sistema eléctrico de manera ideal. Este enfoque es fundamental en GDRTCS porque permite a los simuladores distribuidos mantener la integridad de las señales eléctricas (voltaje y corriente) a través de distancias geográficas, asegurando que la simulación sea lo más realista y precisa posible.

En una simulación distribuida, por ejemplo, el voltaje generado en un subsistema de generación en un simulador puede necesitar ser transmitido a un subsistema de carga en otro simulador ubicado a cientos de kilómetros de distancia. El ITM garantiza que esta transmisión de señales sea efectiva, ajustando y transformando las señales de tal manera que los resultados de la simulación se mantengan consistentes en todo el sistema.

El ITM emplea fuentes de voltaje y corriente controladas para reflejar el comportamiento del subsistema remoto en el subsistema local, garantizando así una interacción coherente y precisa. La estabilidad del sistema se evalúa utilizando el criterio de Nyquist, donde la estabilidad se asegura si la relación de impedancias Z_1/Z_2 se mantiene por debajo de 1.

La ecuación que representa esta relación es:

$$G_0 = \frac{Z_1(s)}{Z_2(s)} e^{-s\Delta t} \quad (1)$$

Donde $Z_1(s)$ representa la impedancia de la red simulada, $Z_2(s)$ la del hardware y $e^{-s\Delta t}$ es el retardo del tiempo.

La implementación y ajuste cuidadoso del ITM es fundamental para garantizar que la simulación representa fielmente las condiciones reales del sistema, proporcionando una herramienta robusta para el desarrollo y evaluación de tecnologías en el ámbito de sistemas eléctricos.

B. Técnica del Modelo de Transformador Ideal Avanzado (AITM)

Una mejora del ITM, que incorpora elementos adicionales para manejar mejor las no-linealidades y mejorar la precisión. El Modelo de Transformador Ideal Avanzado (AITM) es una técnica avanzada utilizada en aplicaciones de Power Hardware-in-the-Loop (PHIL) para mejorar la estabilidad y precisión de la co-simulación. Este modelo emplea:

- Fuentes de voltaje y corriente controladas: Representan el comportamiento de los subsistemas remotos en el subsistema local.
- Impedancias virtuales: Emulan el comportamiento del sistema real, reduciendo discrepancias y mejorando la estabilidad.
- Técnicas de compensación de latencia: Manejan los retardos temporales en la comunicación entre subsistemas, asegurando que las señales sean precisas y sincronizadas.

El AITM mejora la estabilidad y precisión en la co-simulación PHIL mediante fuentes controladas, impedancias virtuales y técnicas de compensación de latencia, proporcionando una plataforma robusta para la simulación de sistemas eléctricos complejos.

Está dado por la siguiente ecuación:

$$G_0 = \frac{Z_1(s)}{Z_c(s) + Z_2(s)} e^{-s\Delta t} \quad (2)$$

Para mejorar la estabilidad del ITM y formar un nuevo AITM, se agrega una impedancia de compensación adicional Z_c al subsistema 1, La Ecuación explica el concepto de AITM, donde Z_c debe ser mayor que 0 para que el AITM evite cortocircuitar la fuente de inyección de

corriente. Se muestran los rangos de impedancia estables del AITM. Solo en $0.5 Z_c$ el sistema entra en una condición inestable cuando $Z_1(s)/Z_2(s)$ supera 1.5. Es importante notar que un componente adicional en el sistema puede alterar la precisión; por lo tanto, debe ser mínimo.

C. Aproximación de Primer Orden Variante en el Tiempo (TFA)

Este método proporciona una mayor flexibilidad y precisión al adaptarse a las variaciones temporales en los parámetros del sistema. Se define como una técnica de análisis utilizada para examinar señales no estacionarias en el dominio del tiempo y la frecuencia simultáneamente. Utiliza métodos como la Transformada de Fourier de Tiempo Corto (STFT) y la Transformada de Wavelet para proporcionar una representación detallada de cómo cambian las características frecuenciales de una señal a lo largo del tiempo [15].

Está dado por las siguientes ecuaciones:

$$\frac{di_2(t)}{dt} = ai_2(t) + bv_2 \quad (3)$$

$$i_2(t) \approx \alpha[v_1(t-1)] + \beta[i_2(t-1)] = \frac{v_1(t-1)}{Z_{eq}} + I_{eq} \quad (4)$$

$$\begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{bmatrix} v_1(t-2) & i_2(t-2) \\ v_1(t-3) & i_2(t-3) \end{bmatrix}^{-1} \begin{bmatrix} i_2(t-1) \\ i_2(t-2) \end{bmatrix} \quad (5)$$

Este método utiliza un enfoque predictivo para estimar cómo se comportará un dispositivo en prueba, empleando sistemas lineales de primer orden. Se guardan y actualizan en tiempo real los datos históricos de las simulaciones, que se usan para ajustar los coeficientes del sistema y del hardware. Así, el simulador puede corregir los errores en la interfaz. En una aplicación típica de un circuito RL, asumiendo que el hardware sigue una ecuación de primer orden, el valor de la corriente i_2 se calcula a partir de los valores iniciales del voltaje v_1 y la corriente i_2 , utilizando una aproximación trapezoidal.

D. Modelo de Línea de Transmisión (TLM)

Utilizado para simular el comportamiento de líneas de transmisión en sistemas eléctricos, especialmente útil en la simulación de fenómenos transitorios. Es una técnica de modelado que representa la propagación de señales en líneas de transmisión a través de una red de nodos interconectados. Se utiliza para simular la interacción entre subsistemas en co-simulación, capturando efectos de propagación, retardo y atenuación de señales.

Está dado por la siguiente ecuación:

$$G_0 = \frac{1 - \alpha e^{-2s\Delta t}}{1 + \alpha e^{-2s\Delta t}} \times \frac{Z_1(s)}{R_t} \quad (6)$$

$$\text{Donde } \alpha = (Z_L - R_t)/(Z_L + R_t)$$

Las conexiones TLM representan componentes reactivos discretos modelados como secciones de líneas de transmisión. Este enfoque considera los componentes reactivos como dispositivos de dos puertos. El cambio en el voltaje del condensador o la corriente del inductor en cada paso de tiempo de la simulación es muy pequeño; por lo tanto, dos modelos de circuito pueden separarse y parecer como si estuvieran conectados a una fuente constante de corriente o voltaje. El método TLM utiliza un inductor o capacitor de enlace con el modelo de línea de Bergeron para reemplazar L o C con R_t . Para reemplazar L con R_t , L se divide por Δt , que es el tiempo de propagación de la línea de transmisión (también el tiempo de retardo de la interfaz), mientras que para reemplazar C con Δt se divide por C. Este método es útil para desacoplar sistemas grandes para permitir cálculos en paralelo, ya que la línea de transmisión puede describirse con el modelo de línea de Bergeron.

E. Duplicación Parcial de Circuito (PCD)

Este algoritmo es adecuado para escenarios donde se necesita replicar parcialmente la dinámica del circuito en la simulación. Es una técnica que divide un circuito complejo en subcircuitos más manejables y encuentra una solución iterativa para mejorar la estabilidad y precisión de la simulación. Utiliza impedancias de acoplamiento adicionales para duplicar enlaces en la red simulada y el hardware de prueba, facilitando una integración estable entre subsistemas.

Está representado por la siguiente ecuación:

$$G_0 = \frac{Z_1(s) Z_2(s)}{[Z_1(s) + Z_{12}(s)][Z_2(s) + Z_{12}(s)]} \times e^{-s\Delta t} \quad (7)$$

El método PCD se basa en la técnica de reducción, que divide el circuito completo en prueba en unidades distintas y se acerca iterativamente a una solución. La ecuación del bucle concluye que la técnica PCD es más estable que el ITM para sistemas resistivos porque es más fácil mantener la función de transferencia por debajo de la unidad. Si esta técnica converge, muchas iteraciones pueden lograr una simulación precisa. Las simulaciones en tiempo real permiten solo una iteración por paso de tiempo. El error de cada iteración debe ser mínimo. Para preservar la precisión, Z_{12} debe ser mayor que Z_1 y Z_2 . En la práctica, esto es difícil de lograr; por lo tanto, la precisión del PCD es generalmente baja. Además, el sistema es estable

incluso con una baja impedancia de acoplamiento Z_{12} . Por lo tanto, agregar impedancias al sistema resulta en un mayor consumo de energía, afectando la fidelidad.

F. Método de Impedancia de Amortiguación (DIM)

Utilizado para mejorar la estabilidad del sistema al proporcionar una impedancia de amortiguación adecuada en la interfaz. El Método de Impedancia de Amortiguamiento (DIM) es una técnica avanzada utilizada para mejorar la estabilidad y precisión de las simulaciones de hardware in the loop (HIL). Este método introduce una resistencia adicional al sistema, conocida como impedancia de amortiguamiento, que actúa como un amortiguador. Su función principal es suavizar las fluctuaciones del sistema, evitando que se vuelva inestable. En esencia, el DIM integra características del Método de Interfaz de Tiempo (ITM) y el Control de Diferencia de Potencia (PCD), utilizando una resistencia de amortiguamiento que se adapta dinámicamente a las condiciones del sistema.

Está dado por la siguiente función de transferencia:

$$G_0 = \frac{Z_1(s)[Z_2(s) - Z^*(s)]}{[Z_1(s) + Z_{12}(s)][Z_2(s) + Z_{12}(s) + Z^*(s)]} \times e^{-s\Delta t} \quad (8)$$

El método DIM incluye una impedancia que optimiza el factor de convergencia, mejorando así la estabilidad y precisión del algoritmo PCD. El DIM es un híbrido de los métodos PCD e ITM, y se logra colocando una impedancia de amortiguación, Z^* , entre las fuentes de corriente y voltaje. La función de transferencia del lazo abierto se presenta en la ecuación correspondiente, y se puede ver que cuando Z^* se aproxima a infinito, el DIM se configura como ITM, y cuando Z^* se aproxima a cero, se configura como PCD. La técnica DIM combina los métodos PCD e ITM usando Z^* . La Figura 10 muestra su implementación. Dependiendo del valor de Z^* , la estabilidad de esta interfaz se sitúa entre la del PCD y la del ITM; por lo tanto, la estabilidad absoluta se garantiza cuando Z^* coincide con Z_2 . Cuando $Z = Z_2$, la función de transferencia G_0 para DIM se convierte en 0, el sistema PHIL es estable y los errores de simulación no se agravarán.

G. Técnica del Filtro de Corriente de Retroalimentación (FCF)

Ayuda a mitigar los efectos de ruido y otras perturbaciones no deseadas en la señal de retroalimentación. El Filtro de Corriente de Retroalimentación (FCF) es una técnica empleada para mejorar la estabilidad de las simulaciones HIL mediante la eliminación del ruido y los

armónicos presentes en la señal de corriente de retroalimentación. Este método implementa un filtro de paso bajo o paso banda, cuyo objetivo es limpiar la señal de corriente antes de utilizarla para controlar el sistema de hardware. Al mejorar la calidad de la señal de corriente, el FCF contribuye significativamente a una simulación más estable y precisa.

Está representado por la siguiente función de transferencia:

$$G_0 = T_{FCF}(s) \frac{Z_1(s)}{Z_2(s)} e^{-s\Delta t} \quad (9)$$

La técnica FCF ayuda a ampliar los rangos de funcionamiento estables del IA al añadir filtros de paso de banda o paso bajo para eliminar armónicos y ruido de la corriente o voltaje de retroalimentación. La ecuación correspondiente muestra la función de transferencia. Esta función de transferencia demuestra la influencia dinámica del filtro de paso bajo en el ITM; de esta manera, agregar un FCF mejora las diferentes frecuencias f_c del ITM. Sin un filtro, el área estable del ITM se ve afectada cuando la proporción supera 1, pero con un FCF de 1000 Hz, la proporción mejora a más de 3. El FCF mejora cada IA, pero puede afectar la precisión del intercambio de señales dependiendo de la frecuencia de corte f_c .

Desafíos y Mejora de la Estabilidad

Los IA no solo deben asegurar la precisión en la transferencia de datos entre los subsistemas, sino también manejar los desafíos inherentes, como el retraso en la red, el ancho de banda limitado y el ruido. Estos desafíos requieren que los IA incluyan técnicas de procesamiento de señales, como filtrado de paso bajo, compensación de ganancia unitaria y compensadores de adelanto-atraso, para mantener la estabilidad y precisión de la simulación.

En consecuencia, la elección y el diseño de un algoritmo de interfaz adecuado es fundamental para la simulación precisa y la gestión eficaz de sistemas eléctricos de potencia, especialmente en procesos avanzados como PHIL y GDRTCS. La óptima y correcta implementación de estos algoritmos permite una mejor representación de las condiciones reales del sistema, lo que es fundamental para el desarrollo y la validación de nuevas tecnologías y estrategias operativas en el campo de los sistemas eléctricos.

Capas Principales

Simulador Digital en Tiempo Real: La capa central de la infraestructura permite un enlace bidireccional punto a punto entre dos simuladores en tiempo real. Los simuladores deben cumplir con:

- Soporta el protocolo Aurora 8B/10B con interfaces SFP para el intercambio de datos entre simuladores en tiempo real.
- Proporcionar una placa de sincronización equipada con una interfaz RJ45 para implementar el protocolo IEEE 1588 Precision Time Protocol (PTP). Esta configuración permite la sincronización de relojes de referencia internos con la capa subyacente.

Sincronización GPS: La capa de sincronización GPS utiliza un reloj GPS para sincronizar su reloj de referencia interno con la tecnología de reloj atómico GPS, creando el nodo maestro IEEE1588 PTP. La infraestructura puede acceder a los estándares de tiempo atómico sin un reloj atómico local, asegurando la sincronización mundial. Dado que los racks DRTS evolucionan su simulación con un esquema de regulación de tiempo independiente, se puede ignorar el esquema de regulación de tiempo. La sección fuente del circuito fuente-carga iniciará la simulación de carga. Las señales GPS aseguran que los subsistemas comienzan sus simulaciones simultáneamente y que los resultados del rack DRTS se alineen utilizando el tiempo del reloj interno.

Co-simulación del Sistema de Energía: Esta capa utiliza el ITM o cualquier IA avanzada para separar un modelo del sistema de energía en dos subsistemas. Este IA, utilizado en aplicaciones PHIL, emplea:

- Un generador de voltaje controlado en el segundo subsistema para recrear el voltaje medido en el primer subsistema.
- Un generador de corriente en el primer subsistema para reproducir la corriente medida en el segundo subsistema. También impone una latencia proporcional a la latencia de la variable transferida desde el primer DRTS al circuito del segundo subsistema y viceversa.

Marco Metodológico

Simulación de algoritmos de interfaz en Simulink

Para llevar a cabo las simulaciones de los algoritmos de interfaz, se diseñaron esquemas que replican de la manera más precisa posible los modelos presentados anteriormente. Estas simulaciones se realizaron utilizando la plataforma Simulink, que ofrece un entorno robusto y flexible para el modelado y la simulación de sistemas dinámicos.

Cada algoritmo de interfaz será implementado y evaluado por separado, permitiendo un análisis detallado de su rendimiento individual. A lo largo de este proceso, se examinarán diversos aspectos como la estabilidad, la precisión y la eficiencia de cada algoritmo en diferentes escenarios y condiciones de prueba.

Estos algoritmos permiten entender, comprender y analizar de manera comparativa los resultados obtenidos cuando se aplican en co-simulaciones utilizando Hardware-in-the-Loop (HIL). Gracias a ellos, se logra una evaluación más precisa y detallada de los sistemas simulados, facilitando la identificación de comportamientos y características clave bajo diferentes escenarios y condiciones de prueba.

Al finalizar las evaluaciones individuales, se llevará a cabo un análisis comparativo exhaustivo. Este análisis comparará las ventajas y desventajas de cada algoritmo, considerando su adecuación a nuestras necesidades específicas. El objetivo es determinar cuál de los algoritmos de interfaz es el más adecuado para nuestra aplicación, basándonos en los criterios de desempeño observados durante las simulaciones y pruebas [16].

VSS (Voltage Source Subsystem)

El VSS (Voltage Source Subsystem) se refiere a la parte del sistema que actúa como la fuente de voltaje en la simulación. Este subsistema es responsable de generar y mantener el nivel de voltaje necesario para alimentar el resto del sistema o para interactuar con otros subsistemas dentro del modelo de simulación. En un entorno de co-simulación, el VSS puede representar, por ejemplo, la generación de energía eléctrica en un punto específico del sistema eléctrico.

PPS (Power Processing Subsystem)

El PPS (Power Processing Subsystem) se refiere a la parte del sistema encargada del procesamiento de la energía, lo que incluye la conversión, regulación y distribución de la energía eléctrica. Este subsistema puede incluir dispositivos como inversores, convertidores, o

unidades de control que modulan la energía suministrada por el VSS para su uso en diferentes partes del sistema eléctrico.

Técnica del Modelo de Transformador Ideal (ITM)

Se realizaron pruebas en simulink para estudiar la respuesta del sistema utilizando el algoritmo ITM, el modelo se representa en la siguiente Figura.

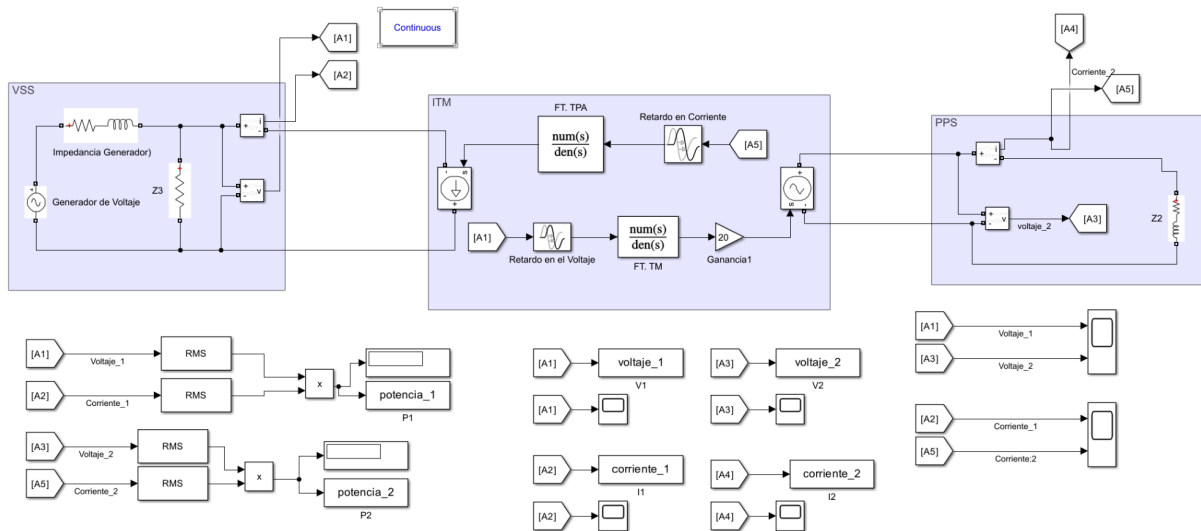


Fig 5. Modelo en Simulink Algoritmo ITM.

Dentro del modelo ITM, el VSS utiliza una fuente de voltaje controlada para imponer el voltaje del subsistema remoto sobre el subsistema local. Esto permite que las variaciones de voltaje en una parte del sistema sean replicadas con precisión en otra parte, garantizando una interacción coherente entre los subsistemas.

En el modelo ITM, el PPS utiliza una fuente de corriente controlada para recibir la señal de corriente desde el subsistema remoto y aplicarla al subsistema local. Esto asegura que las corrientes medidas en un subsistema sean fielmente replicadas en otro, manteniendo la integridad de las simulaciones distribuidas.

Integración en el Modelo ITM

En el modelo ITM implementado, el VSS y el PPS se encuentran interconectados a través de las fuentes de voltaje y corriente controladas que gestionan las señales de intercambio entre ambos subsistemas. Este enfoque permite que la co-simulación sea precisa, reflejando las interacciones dinámicas entre la generación de voltaje y el procesamiento de potencia en un entorno distribuido.

Partes del Modelo con ITM Implementado

Fuentes Controladas de Voltaje y Corriente:

El ITM utiliza fuentes de voltaje y corriente controladas para representar e imponer el comportamiento de un subsistema remoto en un subsistema local.

Estas fuentes controladas permiten que las señales de voltaje y corriente del subsistema remoto sean replicadas de manera precisa en el subsistema local, asegurando que ambos subsistemas interactúen como si estuvieran físicamente conectados.

Función de Transferencia:

Descripción: La función de transferencia del ITM en lazo abierto se define como

$$G_0 = \frac{Z_1(s)}{Z_2(s)} e^{-s\Delta t},$$

donde $Z_1(s)$ es la impedancia del subsistema simulado, $Z_2(s)$ es la impedancia del hardware, y Δt es el retardo total en la interfaz.

Función: Esta función de transferencia es esencial para modelar cómo las señales se transfieren entre los subsistemas

Control de Retardos:

El retardo Δt en la interfaz puede causar desfases y no linealidades que afectan la precisión de la simulación.

Para mitigar estos efectos, se añaden una resistencia en serie y una admitancia en paralelo a las fuentes controladas, lo que ayuda a prevenir la inestabilidad numérica y mantener la coherencia en las señales.

Aplicación en Simulaciones HIL y PHIL:

En entornos HIL y PHIL, el ITM se aplica en la interfaz entre el simulador y el hardware, permitiendo que ambos interactúen de manera precisa y estable.

Este algoritmo asegura que las señales intercambiadas entre el hardware real y la simulación sean coherentes, lo que es crucial para mantener la estabilidad del sistema en tiempo real.

Interconexión de Simuladores Distribuidos

Descripción: En un escenario de simulación distribuida geográficamente, el ITM se utiliza para gestionar la comunicación entre simuladores ubicados en diferentes puntos geográficos.

Función: El algoritmo ITM gestiona el intercambio de voltajes y corrientes entre simuladores, compensando los retardos de transmisión y asegurando que la simulación distribuida funcione de manera unificada y coherente

Para el modelo se utilizaron bloques de Voltaje en AC, impedancias RL para Z_1 y Z_2 . El tiempo de delay para los modelos es de 1 ms. Además de bloques de medición de voltaje y corriente, también un subsistema donde se aplica el algoritmo ITM, con las funciones de transferencia correspondientes, por el otro lado, se utilizó un bloque de Corriente AC, y se procedió a simular el algoritmo.

Se presenta a continuación un pseudocódigo asociado a la implementación de este algoritmo:

1. Inicializar el sistema

- a. Definir las impedancias del sistema:
 - $Z_1(s) = R_1 + sL_1$ // Impedancia del generador (VSS)
 - $Z_2(s) = R_2 + sL_2$ // Impedancia de la carga (PPS)
- b. Configurar las funciones de transferencia para la corriente y el voltaje:
 - $TM(s) = Z_2(s) / (Z_1(s) + Z_2(s))$ // Función de transferencia para la corriente
 - $TPA(s) = Z_1(s) / (Z_1(s) + Z_2(s))$ // Función de transferencia para el voltaje
- c. Establecer la frecuencia de simulación y los parámetros de retardo temporal (si se aplica)

2. Iniciar la simulación

- a. Aplicar una señal de voltaje en el subsistema de la fuente de voltaje (VSS)
 - V_0 = señal de voltaje aplicada en VSS
- b. Calcular la corriente en el subsistema VSS
 - $I'' = V_0 / Z_1(s)$ // Corriente en el subsistema VSS

3. Transformación de la corriente I'' usando $TM(s)$

- a. Aplicar la función de transferencia $TM(s)$
 - $I' = TM(s) * I''$ // Transformación de la corriente
- b. Enviar la corriente transformada I' al subsistema de la carga (PPS)

4. Simulación en el subsistema de la carga (PPS)

- a. Recibir la corriente I' en PPS
- b. Calcular el voltaje en la carga basado en la corriente I'

- $V'' = I' * Z2(s)$ // Voltaje en el subsistema PPS

5. Transformación del voltaje V'' usando TPA(s)

a. Aplicar la función de transferencia TPA(s)

- $V' = \text{TPA}(s) * V''$ // Transformación del voltaje

b. Enviar el voltaje transformado V' de vuelta al subsistema VSS

6. Retroalimentación y ajuste del sistema

a. Recibe el voltaje V' en el subsistema VSS

b. Ajusta la señal de entrada (V_0) o modificar las funciones de transferencia según sea necesario

c. Se repite el ciclo de simulación para analizar el comportamiento en diferentes condiciones

7. Análisis y salida de resultados

a. Ejecuta la simulación para un tiempo específico o hasta la convergencia

b. Recopila los datos de corriente y voltaje en ambos subsistemas (VSS y PPS)

c. Realiza un análisis de Fourier (FFT) en las señales de corriente y voltaje

d. Identifica las componentes de frecuencia, armónicos, y posibles distorsiones

e. Evalúa el rendimiento del algoritmo ITM en la interfaz entre los subsistemas

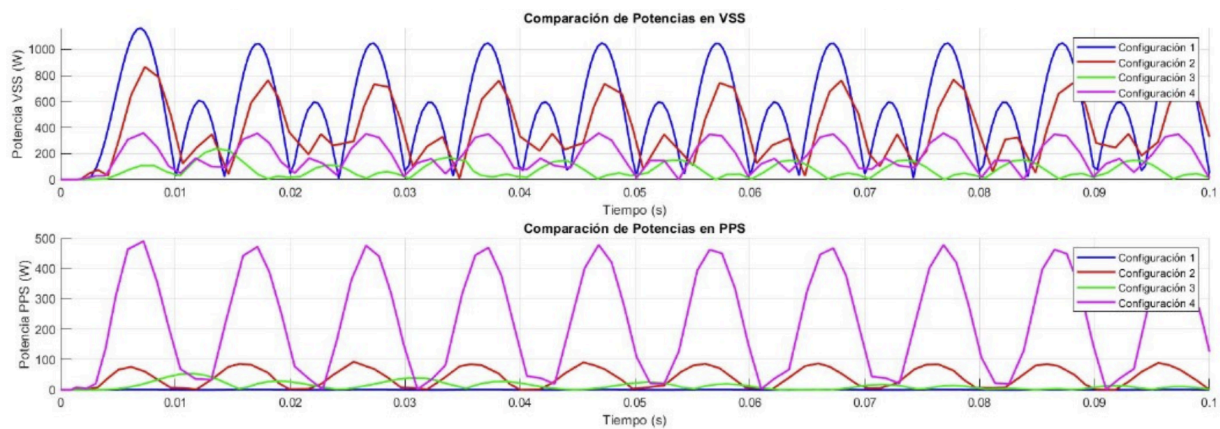


Fig 6. Comparación de Potencias entre el lado VSS y el PPS

El resto de los resultados de la simulación se encuentra en el apartado de Anexos

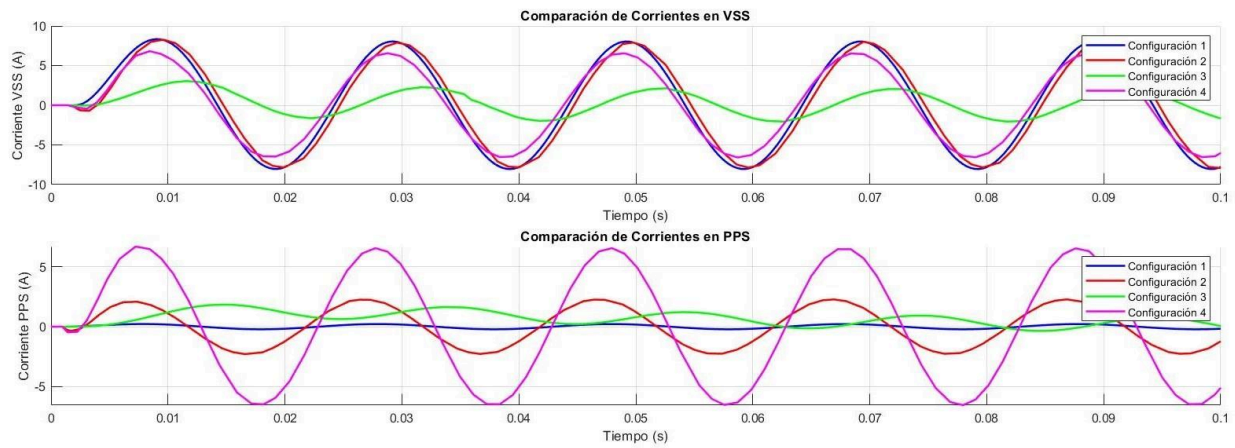


Fig 7. Comparación de Corriente entre el lado VSS y el PPS

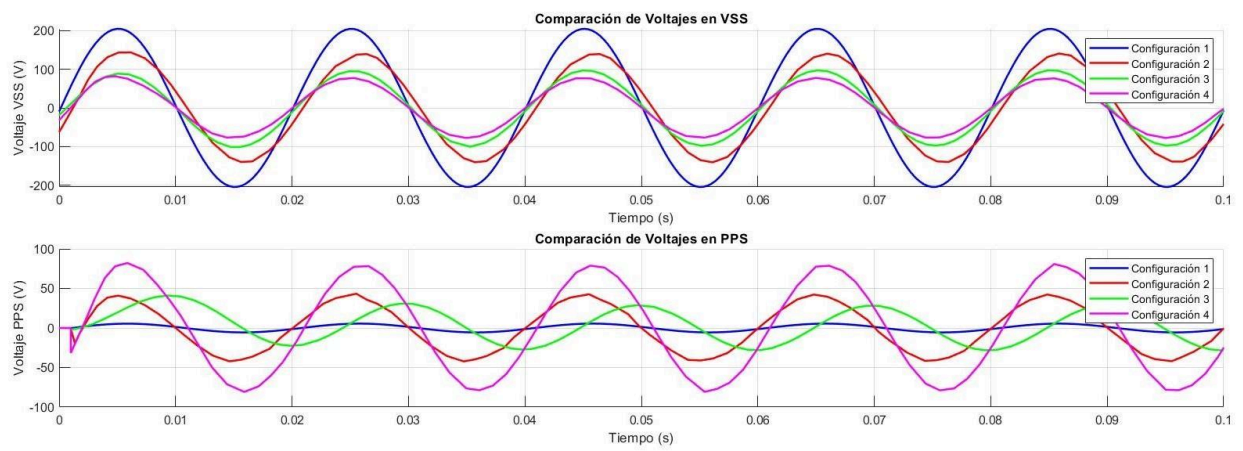


Fig 8. Comparación de Voltaje entre el lado VSS y el PPS

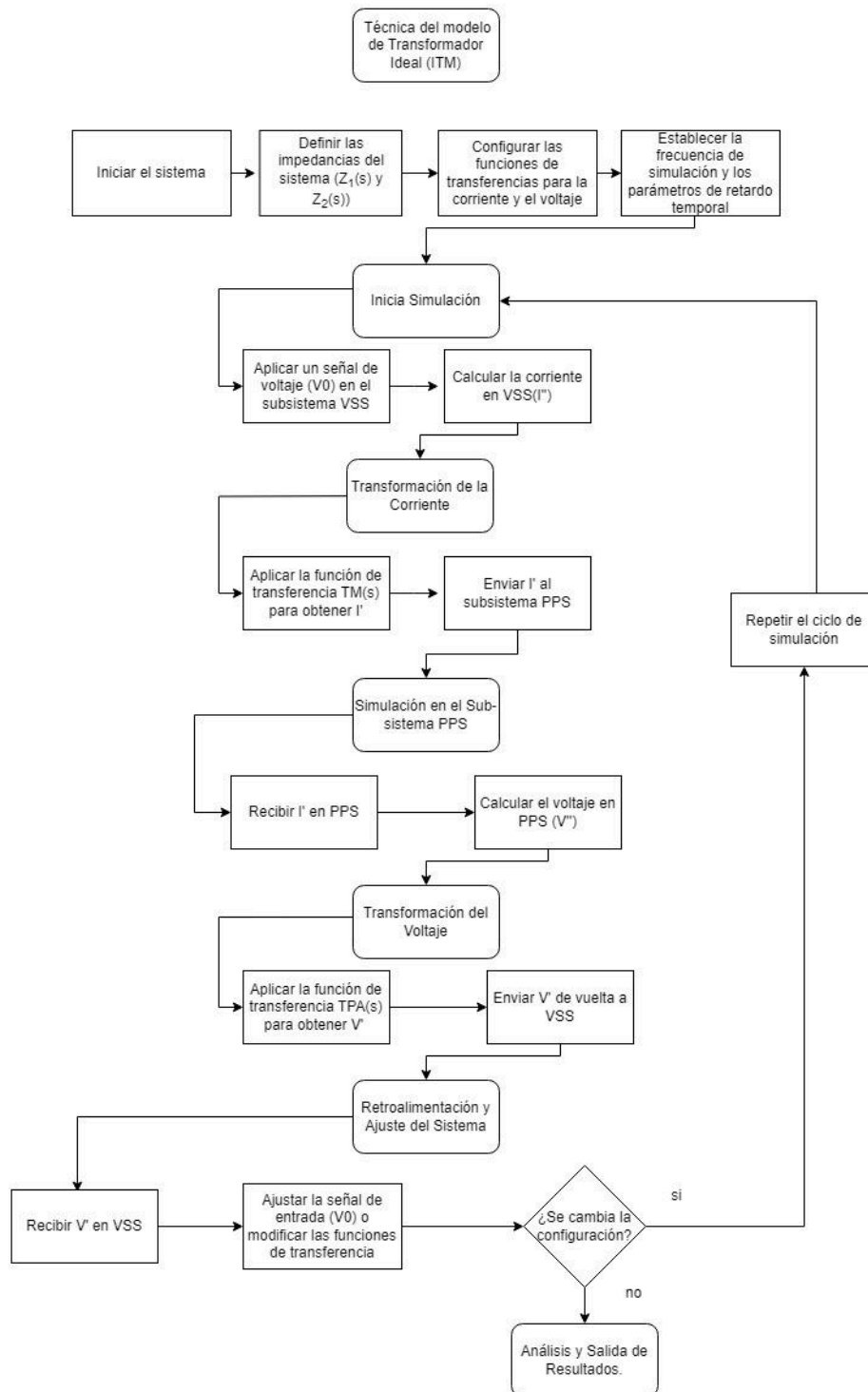


Figura 9. Diagrama de flujo del Algoritmo ITM
Fuente: Elaboración Propia

Técnica del Modelo de Transformador Ideal Avanzado (AITM)

Para la implementación del algoritmo AITM, se le hicieron cambios menores al ITM ya que en su esquema son bastante parecidos.

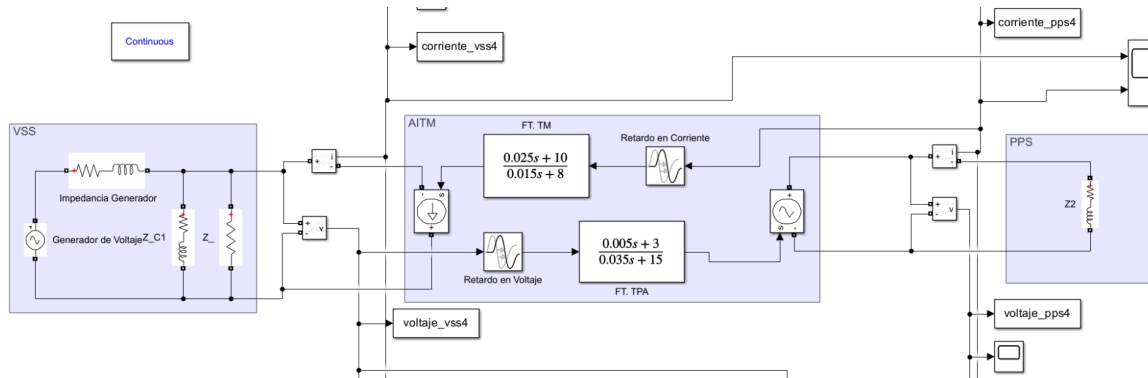


Fig 10. Modelo en Simulink Algoritmo AITM

Como anteriormente mencionado, tiene una similitud considerable, se le agregó un Z_c en paralelo a fuente de corriente controlada, y también se quitó un retardo de tiempo.

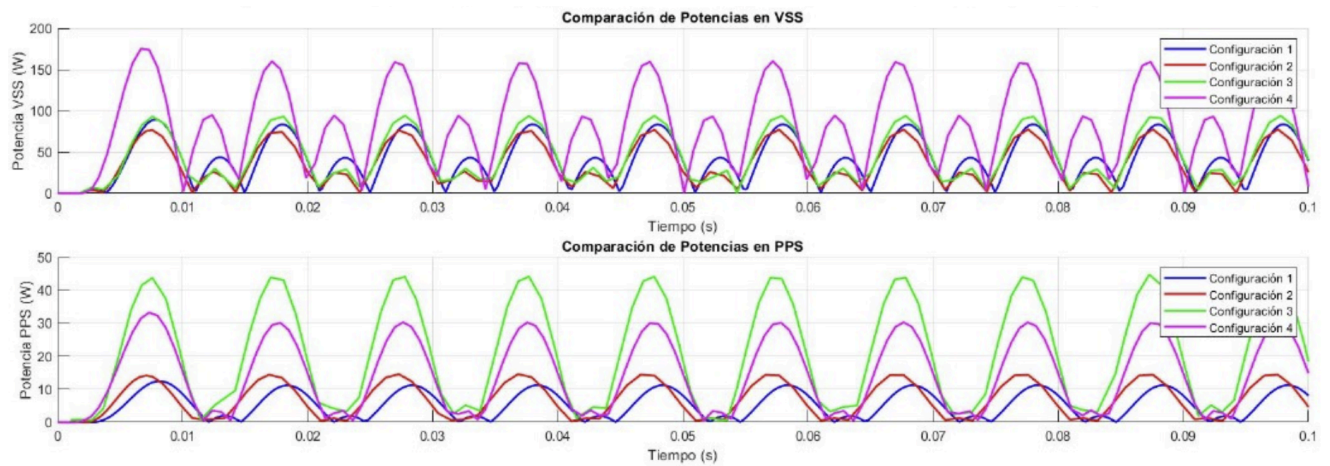


Fig 11. Comparación de Potencias entre el lado VSS y el PPS

El resto de los resultados de la simulación se encuentra en el apartado de Anexos

Modelo de Línea de Transmisión (TLM)

El modelo para el TLM es similar al esquema del PCD, mantiene las mismas características de las fuentes de voltaje controlados, se le agregó un R_t en serie a cada lado del subsistema.

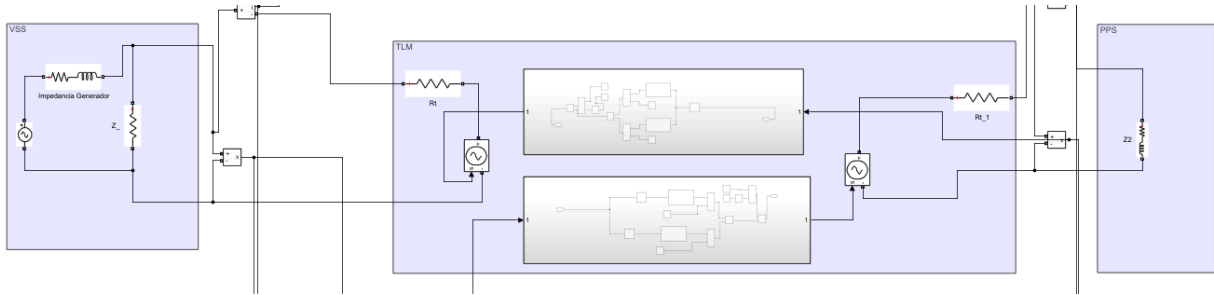


Fig 12. Modelo en Simulink Algoritmo TLM

Duplicación Parcial de Circuito (PCD)

El modelo para el PCD es bastante similar al esquema del AIMT, se le cambia la fuente de corriente controlada por una fuente de voltaje controlada y también se agrega una nueva impedancia Z_{12} en serie a las impedancias anteriores Y el resultado fue el siguiente:

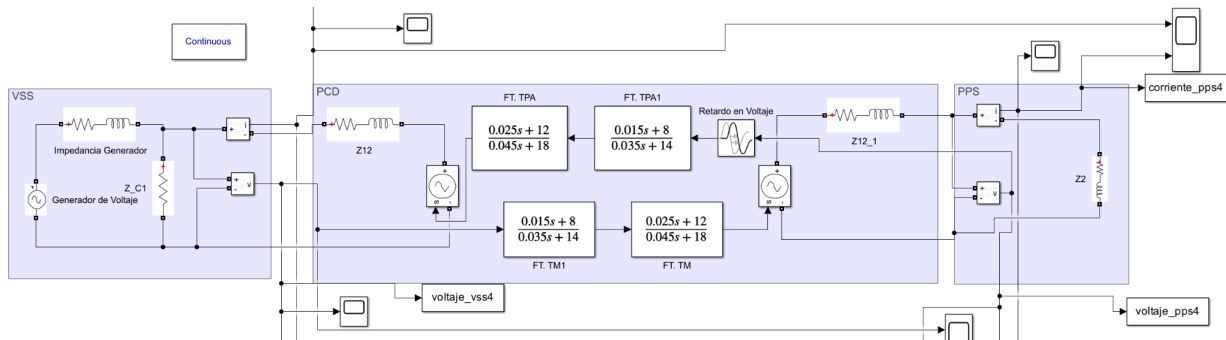


Fig 13. Modelo en Simulink Algoritmo PCD

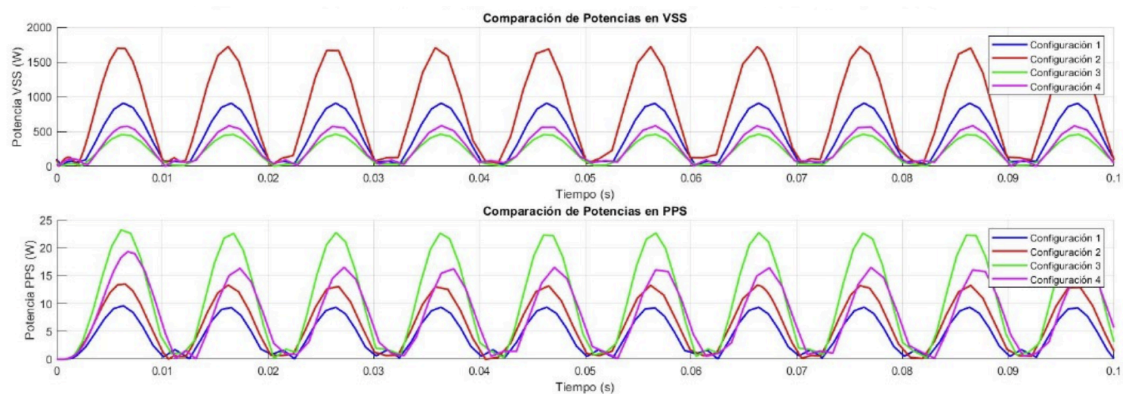


Fig 14. Comparación de Potencias entre el lado VSS y el PPS

Método de Impedancia de Amortiguación (DIM)

El Método de Impedancia de Amortiguación como se mencionó anteriormente, se compone de dos algoritmos de interfaz que son el ITM y PCD, por eso es parecido con los modelos anteriores, ya que contiene las dos fuentes de voltaje controladas y fuente de corriente controlada y además se agregó de una impedancia Z^* de amortiguación en serie al lado del subsistema VSS.

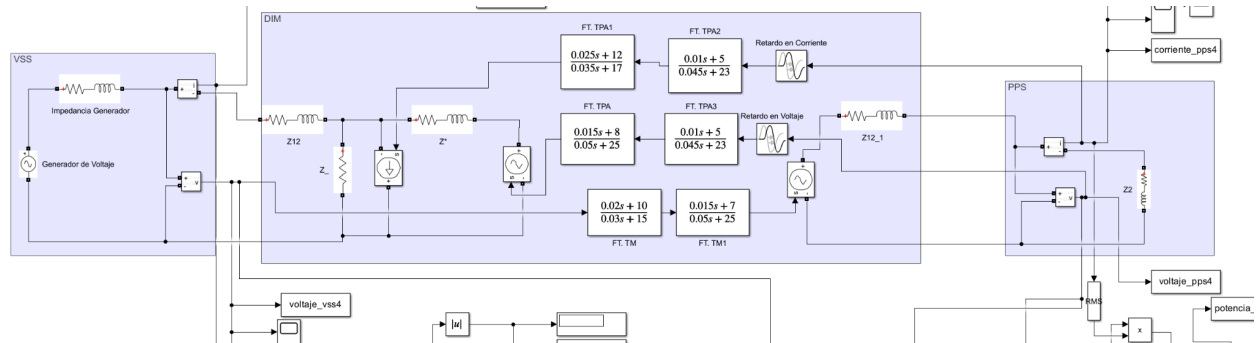


Fig 15. Modelo en Simulink Algoritmo DIM

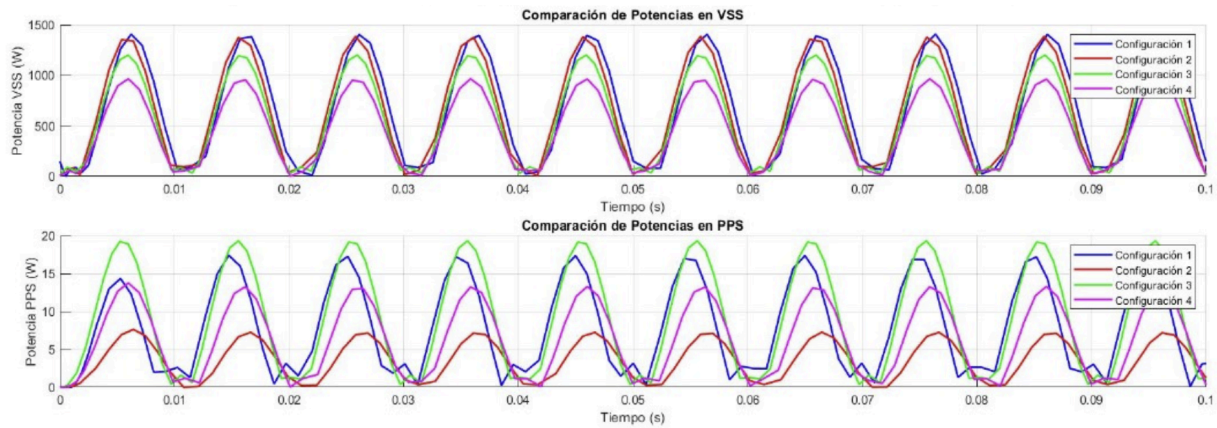


Fig 16. Comparación de Potencias entre el lado VSS y el PPS

Técnica del Filtro de Corriente de Retroalimentación (FCF)

El esquema es bastante parecido a algoritmos anteriores, tiene una parte de ITM, la única diferencia es que el algoritmo contiene un filtro pasa bajo. El algoritmo depende de la frecuencia de corte. El f_c que se usó para el algoritmo es de 100 Hz.

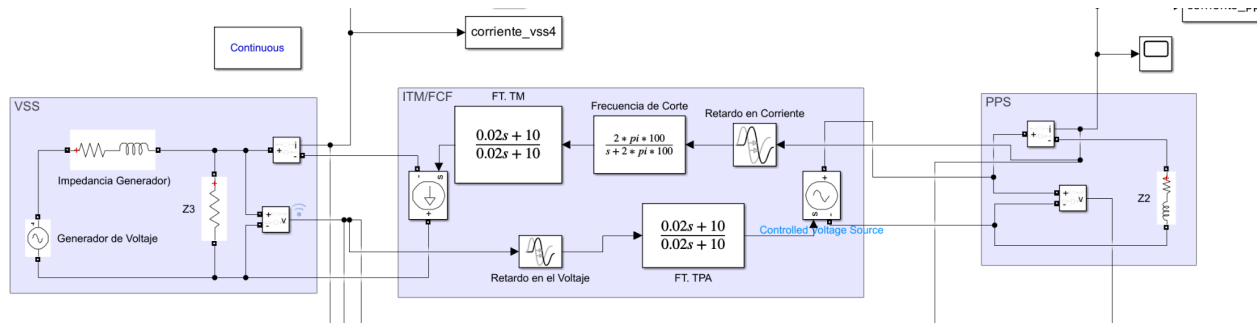


Fig 17. Modelo en Simulink Algoritmo FCF

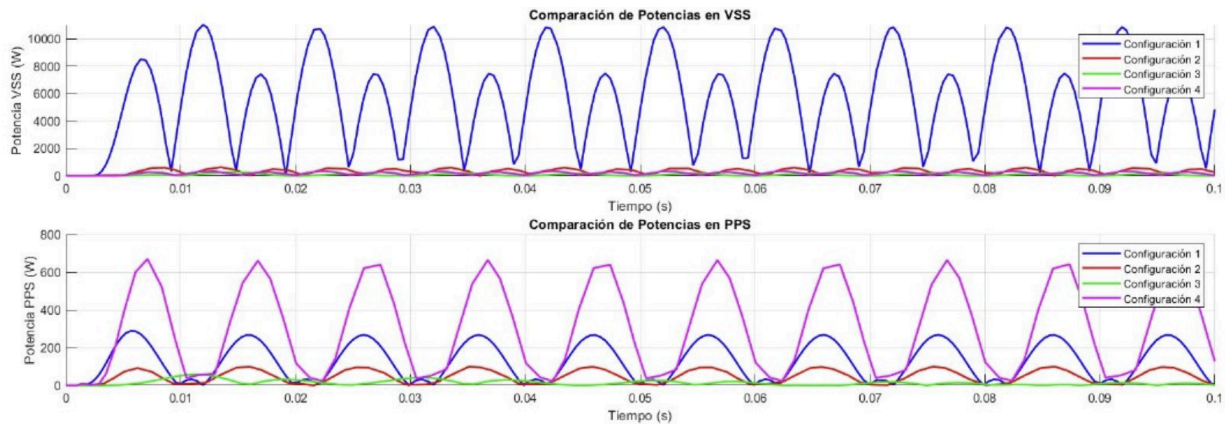


Fig 18. Comparación Potencias del VSS y PPS

Para todos los algoritmos, sólo se mostró la comparación de potencias del VSS y PPS, el resto de los gráficos están en el apartado de Anexos.

Red Federada de Simuladores Distribuidos Geográficamente

Construcción de una Red Federada de Simuladores

En un futuro cercano, se podría construir una red federada de simuladores distribuidos geográficamente que conecte diferentes universidades del país, permitiendo la co-simulación en tiempo real de sistemas eléctricos de potencia a gran escala. Esta red estaría compuesta por una serie de laboratorios ubicados en diversas universidades, cada uno equipado con simuladores avanzados como OPAL-RT y dSPACE.

Por ejemplo, en la Universidad de O'Higgins se podrían instalar simuladores OPAL-RT y dSPACE, mientras que en la Universidad de Chile, específicamente en la Facultad de Ciencias Físicas y Matemáticas (FCFM), se contarían con otros simuladores dedicados. Esta distribución se repetiría en otras universidades del país, como se ilustra en el diagrama de la figura 19, que muestra un mapa donde se ubicarían estos simuladores en cada universidad.

La implementación de esta red federada permitiría que cada universidad se especialice en la simulación de una parte específica del sistema eléctrico de potencia. Por ejemplo, una universidad podría centrarse en la generación de energía, mientras que otra se encargaría de la simulación de la transmisión, y una tercera se enfocaría en la distribución. Esta división no solo reduciría la carga computacional en cada laboratorio, sino que también permitiría un enfoque colaborativo en la investigación y desarrollo de soluciones para sistemas eléctricos complejos.



Figura 19. Mapa de una red federada

Fuente: Elaboración propia

Co-Simulación Distribuida Geográficamente

La conexión de estos simuladores a través de la red federada permitiría la realización de co-simulaciones distribuidas geográficamente. En este esquema, cada simulador actuaría como un nodo en la red, intercambiando datos en tiempo real con otros simuladores ubicados en diferentes universidades. Esta interacción permitiría replicar de manera precisa el comportamiento de un sistema eléctrico a gran escala, donde las diferentes partes del sistema son simuladas en ubicaciones distintas pero interconectadas.

En este contexto, los algoritmos de interfaz como ITM, AITM, TLM, TFA, DIM, PCD y FCF juegan un papel crucial. Estos algoritmos serían implementados en los puntos de conexión entre los simuladores para asegurar la coherencia y estabilidad de la co-simulación. Por ejemplo:

- ITM (Ideal Transformer Model) y AITM (Advanced Ideal Transformer Model) serían utilizados para gestionar la interacción entre las simulaciones de generación y transmisión, asegurando que las señales de corriente y voltaje se transfieran correctamente entre los simuladores.
- TLM (Transmission Line Model) podría ser aplicado en las simulaciones de líneas de transmisión, permitiendo una representación precisa de los retardos y comportamientos dinámicos entre diferentes nodos de la red.
- TFA (Time-Variant First-Order Approximation) y DIM (Damping Impedance Method) serían utilizados para aproximar y estabilizar las señales en sistemas donde se simulan efectos dinámicos complejos, como la integración de energías renovables o la respuesta ante perturbaciones en la red.
- PCD (Partial Circuit Duplication) y FCF (Feedback Current Filter) serían implementados para mejorar la precisión de las simulaciones y mitigar los efectos negativos de los retardos en la comunicación entre simuladores distribuidos.

La correcta implementación y sincronización de estos algoritmos de interfaz es esencial para garantizar que la co-simulación distribuida funcione de manera eficiente y precisa, permitiendo así el análisis detallado y el desarrollo de nuevas tecnologías en el ámbito de los sistemas eléctricos de potencia. Esta red federada no solo fomentará la colaboración entre universidades, sino que también contribuirá al avance en la investigación y la educación en la ingeniería, al proporcionar una plataforma robusta y distribuida para la simulación y el análisis de sistemas eléctricos complejos.

Resultados de los algoritmos implementados

Para llevar a cabo la implementación de los algoritmos en MATLAB, se procede a evaluar cada caso específico utilizando diferentes configuraciones de impedancias que fueron implementadas previamente en Simulink. Se hace un enfoque en los resultados obtenidos y, a partir de estos, se realiza una comparación exhaustiva entre los resultados generados por cada uno de los algoritmos respectivos. Los pseudocódigos de los algoritmos se pueden encontrar en anexos. Se obtuvieron los siguientes resultados en base a las salidas y entradas al VSS de los modelos implementados, los cuales representan el voltaje, corriente y potencia de salida del VSS, y el voltaje, corriente y potencia de entrada del mismo, lo cual permite evaluar y analizar cada algoritmo.

Resultados para el algoritmo ITM

Las pruebas realizadas para el algoritmo ITM se llevaron a cabo con las siguientes configuraciones:

Configuración 1: $Z_1 = 0.5 \Omega + 0.0015H$; $Z_2 = 20 \Omega + 0.05H$

Configuración 2: $Z_1 = 2 \Omega + 0.015H$; $Z_2 = 15 \Omega + 0.03H$

Configuración 3: $Z_1 = 13 \Omega + 0.015H$; $Z_2 = 5 \Omega + 0.15H$

Configuración 4: $Z_1 = 10 \Omega + 0.02H$; $Z_2 = 10 \Omega + 0.02H$

Los resultados obtenidos para cada configuración se evaluaron en términos de las diferencias entre los valores de voltaje, corriente y potencia medidos en los subsistemas VSS y PPS. A continuación, se presentan los resultados estadísticos resumidos en la siguiente tabla:

Tabla 1 Comparación del algoritmo de ITM con diferentes configuraciones.

Configuración	Voltaje (V)	Corriente (A)	Potencia (W)
Configuración 1	Media: -1.5645 Desviación: 322.9949 Máxima: 459.1168 Mínima: -459.0340	Media: 0.1559 Desviación: 5.1469 Máxima: 7.5673 Mínima: -7.3929	Media: 213.6180 Desviación: 67.5532 Máxima: 291.8877 Mínima: -206.9798
Configuración 2	Media: 0.0450 Desviación: 19.7521 Máxima: 28.1749 Mínima: -33.2213	Media: 0.0301 Desviación: 3.6830 Máxima: 5.2543 Mínima: -5.2399	Media: 5.7926 Desviación: 2.5797 Máxima: 6.8143 Mínima: -14.3001
Configuración 3	Media: -0.3102 Desviación: 13.1396 Máxima: 18.5578 Mínima: -25.7851	Media: -0.0317 Desviación: 0.1877 Máxima: 0.2313 Mínima: -0.6556	Media: 0.0735 Desviación: 0.7925 Máxima: 0.7406 Mínima: -2.5648
Configuración 4	Media: -0.0956 Desviación: 3.7497 Máxima: 5.5612 Mínima: -8.6136	Media: 0.0141 Desviación: 0.3065 Máxima: 0.4448 Mínima: -0.4445	Media: -0.0788 Desviación: 0.3182 Máxima: 0.5139 Mínima: -1.3269

Análisis de Resultados ITM

Los resultados obtenidos muestran una variabilidad significativa en el comportamiento del algoritmo ITM bajo diferentes configuraciones. Las configuraciones con impedancias más elevadas en el subsistema VSS tienden a presentar mayores diferencias en el voltaje y la potencia, lo que indica una mayor sensibilidad del algoritmo a la relación de impedancias entre los subsistemas. Sin embargo, en configuraciones con impedancias más equilibradas, como en la cuarta configuración, se observa una mayor estabilidad y coherencia en los valores de voltaje y corriente, lo que sugiere que el ITM puede ser particularmente efectivo cuando las impedancias de los subsistemas están adecuadamente ajustadas.

Resultados para el algoritmo AITM

El Advanced Ideal Transformer Model (AITM) se implementó para mejorar la estabilidad y precisión del algoritmo ITM añadiendo una impedancia de compensación Z_c al subsistema 1. Para evaluar el rendimiento del AITM, se definieron cuatro configuraciones diferentes, considerando las impedancias Z_1, Z_2 y Z_c tanto para el subsistema de origen (VSS) como para el de destino (PPS).

Configuración 1: $Z_1 = 4 \Omega + 0.01H$; $Z_2 = 16 \Omega + 0.05H$; $Z_c = 8 \Omega + 0.015H$

Configuración 2: $Z_1 = 5 \Omega + 0.015H$; $Z_2 = 20 \Omega + 0.04H$; $Z_c = 10 \Omega + 0.02H$

Configuración 3: $Z_1 = 6 \Omega + 0.02H$; $Z_2 = 12 \Omega + 0.03H$; $Z_c = 7 \Omega + 0.01H$

Configuración 4: $Z_1 = 3 \Omega + 0.005H$; $Z_2 = 10 \Omega + 0.025H$; $Z_c = 5 \Omega + 0.01H$

Los resultados obtenidos, que miden las diferencias entre los valores de voltaje, corriente y potencia entre los subsistemas VSS y PPS, se resumen en la tabla que estará en el apartado en anexos.

Análisis de Resultados AITM

Los resultados del AITM muestran que la inclusión de una impedancia de compensación Z_c influye significativamente en la estabilidad y precisión del algoritmo. En la Configuración 1, que tiene impedancias más balanceadas entre Z_1 , Z_2 y Z_c , se observa una menor desviación estándar en las diferencias de voltaje y corriente, lo que indica una mayor estabilidad del sistema. Por otro lado, la Configuración 3 muestra una desviación considerable en la potencia, sugiriendo que cuando Z_c es menor en comparación con Z_1 y Z_2 , el sistema puede volverse menos estable, reflejando una mayor variabilidad en los resultados.

La Configuración 4 presenta una mayor desviación estándar en el voltaje y la corriente, lo que sugiere que la elección de Z_c es crítica para mantener la estabilidad del sistema en el AIM. Este comportamiento destaca la importancia de ajustar adecuadamente las impedancias de compensación para asegurar un desempeño óptimo del algoritmo.

Estos resultados subrayan la efectividad del AIM para mejorar la estabilidad y precisión en sistemas con impedancias adecuadamente balanceadas. Sin embargo, también evidencian que una configuración inadecuada de Z_c puede resultar en una menor estabilidad, lo cual es crucial a considerar en aplicaciones de co-simulación y HIL.

Resultados para el Algoritmo TLM

El Transmission Line Model (TLM) es un algoritmo de interfaz que modela componentes reactivos discretos como secciones de líneas de transmisión. Este enfoque es particularmente útil para desacoplar grandes sistemas y facilitar el cálculo paralelo en simulaciones distribuidas. Para evaluar el desempeño del TLM, se implementaron cuatro configuraciones diferentes, considerando las impedancias Z_1 y Z_2 en los subsistemas VSS y PPS, respectivamente, así como la impedancia de la línea de transmisión R_t y el retardo en la transmisión $\Delta t = 1 \text{ ms}$.

Configuración 1: $Z_1 = 4 \Omega + 0.01H$; $Z_2 = 16 \Omega + 0.05H$; $R_t = 8 \Omega$

Configuración 2: $Z_1 = 5 \Omega + 0.015H$; $Z_2 = 20 \Omega + 0.04H$; $R_t = 10 \Omega$

Configuración 3: $Z_1 = 6 \Omega + 0.02H$; $Z_2 = 12 \Omega + 0.03H$; $R_t = 12 \Omega$

Configuración 4: $Z_1 = 8 \Omega + 0.015H$; $Z_2 = 10 \Omega + 0.02H$; $R_t = 15 \Omega$

Los resultados obtenidos, que miden las diferencias entre los valores de voltaje, corriente y potencia entre los subsistemas VSS y PPS, se resumen en la tabla que estará en el apartado en anexos.

Análisis de Resultados TLM

Los resultados del TLM muestran que este modelo proporciona una excelente precisión en la simulación de sistemas eléctricos, con pequeñas diferencias entre los valores de voltaje, corriente y potencia en los subsistemas VSS y PPS. En la Configuración 1, se observa una mínima diferencia de voltaje y corriente entre los subsistemas, lo que indica que la configuración del TLM es capaz de mantener una alta precisión en la transferencia de señales.

La Configuración 4, que presenta la mayor impedancia de línea de transmisión Rt y un tiempo de retardo de 1ms, muestra valores de diferencia ligeramente más altos en comparación con las otras configuraciones. Sin embargo, estas diferencias siguen siendo pequeñas, lo que subraya la efectividad del TLM en la preservación de la coherencia entre los subsistemas.

En general, los resultados confirman que el TLM es un algoritmo adecuado para aplicaciones que requieren alta precisión y estabilidad en la simulación distribuida de sistemas eléctricos de potencia. La capacidad del TLM para modelar los efectos de retardo en la transmisión y mantener la integridad de la señal lo hace ideal para su uso en co-simulaciones que involucran componentes distribuidos geográficamente.

Resultados para el Algoritmo PCD

El Partial Circuit Duplication (PCD) es un algoritmo de interfaz que se utiliza para desacoplar sistemas eléctricos en la co-simulación, dividiendo el circuito original en subcircuits que se interconectan mediante impedancias de acoplamiento adicionales. Este enfoque permite que las simulaciones en tiempo real mantengan la estabilidad y precisión, especialmente en sistemas donde la interacción entre subsistemas es crítica.

Para evaluar el rendimiento del PCD, se realizaron pruebas con cuatro configuraciones diferentes de impedancias Z_1 , Z_2 , y Z_{12} en los subsistemas VSS y PPS. Estas configuraciones se diseñaron para analizar el impacto de diferentes niveles de acoplamiento entre los subsistemas.

Configuración 1: $Z_1 = 5 \Omega + 0.01H$; $Z_2 = 15 \Omega + 0.05H$; $Z_{12} = 10 \Omega + 0.02H$

Configuración 2: $Z_1 = 3 \Omega + 0.005H$; $Z_2 = 20 \Omega + 0.03H$; $Z_{12} = 8 \Omega + 0.015H$

Configuración 3: $Z_1 = 10 \Omega + 0.02H$; $Z_2 = 10 \Omega + 0.02H$; $Z_{12} = 5 \Omega + 0.01H$

Configuración 4: $Z_1 = 8 \Omega + 0.015H$; $Z_2 = 12 \Omega + 0.025H$; $Z_{12} = 6 \Omega + 0.02H$

Los resultados obtenidos, que miden las diferencias entre los valores de voltaje, corriente y potencia entre los subsistemas VSS y PPS, se resumen en la tabla que estará en el apartado en anexos.

Análisis de Resultados PCD

Los resultados del algoritmo PCD muestran diferencias significativas en el voltaje, corriente y potencia entre los subsistemas VSS y PPS. Estas diferencias reflejan el impacto de las

configuraciones de las impedancias de acoplamiento Z_{12} en la estabilidad y precisión del sistema.

En la Configuración 1, se observan grandes diferencias en voltaje y corriente, lo que sugiere que la impedancia de acoplamiento utilizada en esta configuración no es lo suficientemente alta para garantizar la estabilidad del sistema. La Configuración 2 presenta las mayores diferencias, lo que indica una posible inestabilidad en la simulación, atribuible a la combinación de impedancias seleccionadas.

En contraste, la Configuración 3 muestra una mejor estabilidad con diferencias más bajas en corriente y potencia, aunque todavía presenta variabilidad en el voltaje. Finalmente, la Configuración 4 ofrece un balance intermedio, pero con una ligera mejora en la estabilidad respecto a la Configuración 2.

En general, el análisis sugiere que el PCD es más sensible a las configuraciones de impedancia, especialmente en relación con Z_{12} . La correcta selección de estas impedancias es crucial para garantizar la precisión y estabilidad de la co-simulación, destacando la importancia de un cuidadoso ajuste de los parámetros en este algoritmo.

Resultados para el Algoritmo DIM

El Damping Impedance Method (DIM) es un algoritmo de interfaz que combina las características del ITM (Ideal Transformer Model) y el PCD (Partial Circuit Duplication), introduciendo una impedancia de amortiguamiento Z^* entre las fuentes de corriente y voltaje controladas. Esta técnica busca optimizar la estabilidad y precisión del sistema al ajustar la impedancia de amortiguamiento en función de las condiciones del sistema.

Para evaluar el rendimiento del DIM, se realizaron pruebas con cuatro configuraciones diferentes de impedancias Z_1 , Z_2 , Z_{12} y Z^* en los subsistemas VSS y PPS. Estas configuraciones fueron diseñadas para analizar cómo varía la estabilidad del sistema con diferentes valores de Z^* .

Configuración 1: $Z_1 = 4 \Omega + 0.02H$; $Z_2 = 15 \Omega + 0.05H$; $Z^* = 6 \Omega + 0.01H$
 $Z_{12} = 10 \Omega + 0.02H$

Configuración 2: $Z_1 = 5 \Omega + 0.01H$; $Z_2 = 20 \Omega + 0.03H$; $Z^* = 7 \Omega + 0.015H$
 $Z_{12} = 12 \Omega + 0.025H$

Configuración 3: $Z_1 = 8 \Omega + 0.015H$; $Z_2 = 4 \Omega + 0.005H$; $Z^* = 6 \Omega + 0.015H$
 $Z_{12} = 5 \Omega + 0.01H$

Configuración 4: $Z_1 = 10 \Omega + 0.02H$; $Z_2 = 12 \Omega + 0.025H$; $Z^* = 5 \Omega + 0.01H$
 $Z_{12} = 8 \Omega + 0.015H$

Los resultados obtenidos, que miden las diferencias entre los valores de voltaje, corriente y potencia entre los subsistemas VSS y PPS, se resumen en la tabla que estará en el apartado en anexos.

Análisis de Resultados

Los resultados del algoritmo DIM muestran variaciones en las diferencias de voltaje, corriente y potencia entre los subsistemas VSS y PPS en función de las configuraciones de las impedancias. Estas diferencias indican cómo el ajuste de la impedancia de amortiguamiento Z^* puede afectar la estabilidad y precisión del sistema.

En la Configuración 1, se observan grandes diferencias en voltaje, lo que sugiere una alta variabilidad en la respuesta del sistema. La Configuración 2, aunque presenta una media de diferencia negativa en voltaje y corriente, muestra una mejor estabilidad general en comparación con la Configuración 1.

La Configuración 3 presenta las diferencias más bajas en voltaje, corriente y potencia, indicando una configuración más equilibrada y estable. Finalmente, la Configuración 4 muestra una mejora significativa en la estabilidad de la potencia, aunque con una ligera variabilidad en el voltaje y la corriente.

Como conclusión, se sugiere que el DIM es efectivo para mejorar la estabilidad del sistema, pero su rendimiento depende en gran medida de la selección adecuada de la impedancia de amortiguamiento Z^*

Resultados para el Algoritmo FCF

El Feedback Current Filter (FCF) es un algoritmo de interfaz diseñado para mejorar la estabilidad de otros algoritmos de interfaz, como el ITM, mediante la adición de un filtro de corriente de retroalimentación. Este filtro permite eliminar armónicos y ruido de la señal de

corriente o voltaje, extendiendo así el rango de operación estable del sistema y mejorando la precisión de la simulación.

Para evaluar el rendimiento del FCF, se realizaron pruebas con cuatro configuraciones diferentes de impedancias Z_1 y Z_2 , utilizando un filtro de corte a 100 Hz. Estas configuraciones fueron seleccionadas para observar cómo afecta el filtro de retroalimentación a la estabilidad y precisión del sistema en diferentes escenarios.

Las configuraciones son las mismas que se usaron en ITM, ya que el filtro para este caso está actuando en el algoritmo ITM

Configuración 1: $Z_1 = 0.5 \Omega + 0.0015H$; $Z_2 = 20 \Omega + 0.05H$

Configuración 2: $Z_1 = 2 \Omega + 0.015H$; $Z_2 = 15 \Omega + 0.03H$

Configuración 3: $Z_1 = 13 \Omega + 0.015H$; $Z_2 = 5 \Omega + 0.15H$

Configuración 4: $Z_1 = 10 \Omega + 0.02H$; $Z_2 = 10 \Omega + 0.02H$

Los resultados obtenidos, que miden las diferencias entre los valores de voltaje, corriente y potencia entre los subsistemas VSS y PPS, se resumen en la tabla que estará en el apartado en anexos.

Análisis de Resultados FCF

Los resultados del algoritmo FCF muestran cómo la adición de un filtro de retroalimentación afecta las diferencias de voltaje, corriente y potencia entre los subsistemas VSS y PPS en distintas configuraciones de impedancias. Este análisis es crucial para entender el impacto del FCF en la estabilidad y precisión de la simulación.

En la Configuración 1, la media de la diferencia de voltaje es positiva, pero se observa una desviación estándar considerable en los valores de voltaje y corriente, lo que indica cierta variabilidad en la respuesta del sistema. La Configuración 2 presenta una media de diferencia de voltaje más alta y una desviación estándar similar, lo que sugiere que el filtro está afectando de manera significativa la respuesta del sistema.

La Configuración 3 muestra una media de diferencia de voltaje negativa y una desviación estándar de voltaje considerablemente alta, lo que indica una variabilidad extrema en el sistema. Esto podría ser resultado de un filtrado inadecuado o de una configuración de impedancia que no es óptima para este tipo de filtro.

Finalmente, la Configuración 4 muestra una media de diferencia de voltaje cercana a cero y una desviación estándar más baja en comparación con las configuraciones anteriores, lo que sugiere que el sistema es más estable con esta configuración. Sin embargo, la diferencia de potencia es negativa, lo que indica una posible pérdida de eficiencia en la transmisión de energía entre los subsistemas.

Dentro del análisis se sugiere que el FCF puede mejorar la estabilidad del sistema, pero su efectividad depende de la correcta configuración de las impedancias y del filtro utilizado. Las configuraciones con desviaciones estándar más bajas en voltaje y corriente tienden a ofrecer una mayor estabilidad y precisión en la simulación, mientras que las configuraciones con alta variabilidad pueden indicar problemas en la implementación del filtro o en las condiciones del sistema.

Discusión y Análisis

La implementación y comparación de los algoritmos de interfaz (IA) dentro del contexto de la co-simulación en sistemas eléctricos de potencia nos han permitido identificar las fortalezas y debilidades de cada técnica, así como su impacto en la precisión, estabilidad y eficiencia computacional. En particular, los algoritmos ITM, AITM, TLM, PCD, DIM y FCF fueron evaluados bajo diversas configuraciones de impedancias y condiciones de simulación, lo que nos brinda una visión integral sobre su comportamiento en escenarios de co-simulación distribuida.

1. Análisis Comparativo de los Algoritmos

ITM (Ideal Transformer Model):

El ITM es un algoritmo sencillo y ampliamente utilizado que permite imponer el comportamiento de un subsistema remoto sobre otro subsistema local. A lo largo de las pruebas realizadas, se observó que el ITM ofrece una buena estabilidad y precisión cuando las impedancias están bien configuradas. Sin embargo, presenta limitaciones en términos de retardo y errores en la señal cuando la distancia y la latencia entre subsistemas son significativas. Esto puede afectar la sincronización y la coherencia de la simulación, especialmente en sistemas distribuidos geográficamente.

AITM (Advanced Ideal Transformer Model):

El AITM es una extensión del ITM que mejora su estabilidad mediante la adición de una impedancia de compensación. Los resultados mostraron que, en general, el AITM ofrece una mayor estabilidad en comparación con el ITM, reduciendo la variabilidad en la diferencia de voltajes y corrientes entre los subsistemas VSS y PPS. No obstante, la efectividad del AITM depende en gran medida de la correcta selección de la impedancia de compensación, lo que puede requerir un ajuste fino para lograr un rendimiento óptimo.

TLM (Transmission Line Model):

El TLM es particularmente útil para sistemas que incluyen elementos reactivos modelados como secciones de línea de transmisión. Las pruebas realizadas mostraron que el TLM es eficaz en la decopulación de sistemas grandes, permitiendo una simulación más estable y paralela. Sin embargo, este algoritmo también está sujeto a los efectos de retardo debido a la propagación de la línea de transmisión, lo que puede introducir desfases en la simulación si no se manejan adecuadamente.

PCD (Partial Circuit Duplication):

El PCD se enfoca en dividir el circuito original en subcircuitos y realizar iteraciones para alcanzar una solución. Aunque ofrece una estabilidad mejorada en sistemas resistivos, se observó que su precisión puede ser baja si la impedancia de acoplamiento no se dimensiona adecuadamente. En las configuraciones probadas, el PCD mostró una mayor variabilidad en las diferencias de voltaje y corriente, lo que sugiere que podría ser menos adecuado para sistemas con alta complejidad reactiva o inductiva.

DIM (Damping Impedance Method):

El DIM es una técnica híbrida que combina elementos del PCD y del ITM, introduciendo una impedancia de amortiguamiento para mejorar la estabilidad y precisión. Los resultados mostraron que el DIM es eficaz en la mejora de la estabilidad del sistema, particularmente en escenarios donde las impedancias son variables. Sin embargo, su implementación requiere una cuidadosa selección de la impedancia de amortiguamiento para evitar efectos negativos en la precisión de la simulación.

FCF (Feedback Current Filter):

El FCF tiene como objetivo mejorar la estabilidad del ITM mediante la eliminación de armónicos y ruido en la señal de corriente o voltaje. Los resultados mostraron que el FCF puede mejorar significativamente la estabilidad en ciertas configuraciones, aunque en algunas pruebas se observó una variabilidad considerable, lo que sugiere que la efectividad del filtro depende en gran medida de la configuración específica del sistema. Esto lo hace ideal para situaciones donde se busca extender el rango operativo estable, pero puede requerir ajustes adicionales para minimizar la distorsión de la señal.

2. Implicaciones para la Co-Simulación Distribuida

La co-simulación distribuida en sistemas eléctricos de potencia plantea desafíos significativos, especialmente en términos de latencia y sincronización entre simuladores geográficamente dispersos. Los algoritmos de interfaz juegan un papel crucial en la mitigación de estos desafíos, permitiendo la integración efectiva de subsistemas que operan en diferentes ubicaciones.

Precisión y Estabilidad:

Los algoritmos ITM, AITM y TLM, aunque efectivos en términos de estabilidad, muestran limitaciones en escenarios con alta latencia, lo que puede afectar la precisión de la simulación. Por otro lado, el PCD y el DIM ofrecen soluciones más estables, pero con un mayor requerimiento de ajuste en sus configuraciones.

Eficiencia Computacional:

El TLM destaca en términos de eficiencia computacional al permitir la paralelización de grandes sistemas. Sin embargo, el aumento en la complejidad de la configuración puede requerir recursos adicionales para ajustar los parámetros de las líneas de transmisión.

Impacto del Retardo:

Todos los algoritmos son susceptibles a los efectos del retardo en la comunicación entre simuladores distribuidos, lo que puede introducir errores en la simulación. El uso de técnicas como el FCF puede ayudar a mitigar algunos de estos efectos, pero su implementación requiere un balance cuidadoso entre estabilidad y precisión.

Tabla 2. Comparativa de Algoritmos de interfaz, basado en [17].

Algoritmos	Ventajas	Desventajas
ITM	Simple, buena precisión, mínima computación	Estabilidad baja, retraso de tiempo puede causar inestabilidad
AITM	Estabilidad baja, retraso de tiempo puede causar inestabilidad	La impedancia adicional puede alterar la precisión
TFA	Método predictivo, puede corregir errores de interfaz	Requiere datos históricos y actualizaciones en tiempo real
TLM	Útil para desacoplar sistemas grandes, permite computación en paralelo	Requiere modelos de línea de transmisión
PCD	Más estable para sistemas resistivos, permite iteraciones múltiples	Precisión baja debido a la dificultad de mantener Z_{12} mayor que Z_1 y Z_2
DIM	Optimiza el factor de convergencia, combina PCD e ITM	La estimación de Z^* puede ser difícil, depende de un modelo de hardware adecuado
FCF	Extiende rangos de funcionamiento estable, elimina armónicos y ruido	Puede afectar la precisión del intercambio de señales dependiendo de la frecuencia de corte f_C .

Uso de equipos OPAL-RT para HIL

Durante las pruebas realizadas con los simuladores OPAL-RT, se utilizó el OPAL 4512 como planta y el OPAL 4200 como controlador para ejecutar simulaciones en tiempo real (RTS) y en hardware-in-the-loop (HIL). Sin embargo, debido a problemas técnicos, no se pudieron continuar las pruebas ni se llevó a cabo la co-simulación planificada. Ambos simuladores estaban conectados dentro del mismo laboratorio, lo que limitó la posibilidad de explorar escenarios más complejos de co-simulación con simuladores geográficamente distribuidos. A pesar de estas limitaciones, los resultados obtenidos subrayan la necesidad de realizar futuros estudios con los simuladores ubicados en diferentes entornos, lo que permitirá un análisis más profundo de la co-simulación y sentará las bases para la implementación de una red federada de simuladores.

Conclusiones

En esta memoria se ha realizado un análisis exhaustivo de los protocolos de comunicación y los algoritmos de interfaz aplicados en la co-simulación de sistemas eléctricos de potencia, con un enfoque particular en el contexto de Power Hardware-in-the-Loop (PHIL) y la Co-Simulación en Tiempo Real Distribuida Geográficamente (GDRTCS). Se ha demostrado cómo estos elementos son fundamentales para la interoperabilidad y la gestión eficiente de sistemas complejos, especialmente en entornos donde la integración de energías renovables y la automatización son cada vez más críticas.

Los protocolos de comunicación, como IEC 61850, Modbus, DNP3 e IEC 60870-5, permiten un intercambio de información preciso y confiable, asegurando la operación segura y eficiente de la red eléctrica. Estos protocolos son esenciales para la coordinación y el control de los sistemas eléctricos, facilitando la integración de diversos equipos y tecnologías en un entorno de Smart Grids.

Por otro lado, los algoritmos de interfaz como ITM, AITM, FCF, DIM, y PCD juegan un papel clave en la co-simulación al permitir la interacción precisa entre subsistemas físicos y virtuales. Cada algoritmo ofrece ventajas y desventajas específicas: mientras que ITM, FCF y AITM son efectivos en la atenuación de señales no deseadas, pueden comprometer la cantidad de información útil conservada. En contraste, DIM y PCD proporcionan una mejor preservación de la señal original, siendo más adecuados para aplicaciones que requieren alta fidelidad.

Las pruebas realizadas con las configuraciones de estos algoritmos han mostrado que la selección adecuada de cada uno depende del objetivo específico del sistema a simular. La estabilidad de la simulación y la eficiencia en la transmisión de potencia son factores críticos que deben ser cuidadosamente considerados. Los resultados obtenidos subrayan la necesidad de ajustar y optimizar estos algoritmos para mejorar la precisión y la estabilidad de las simulaciones, contribuyendo a una gestión más segura y eficiente de los recursos eléctricos.

Las pruebas iniciales con los simuladores OPAL 4512 y OPAL 4200 han mostrado el potencial de las simulaciones en tiempo real y HIL para abordar los desafíos complejos de la ingeniería eléctrica. Aunque la co-simulación no se realizó debido a limitaciones técnicas y de proximidad física, estos estudios destacan la importancia de continuar investigando con simuladores separados geográficamente. En el futuro, estas investigaciones podrían facilitar la creación de una red federada de simuladores, permitiendo la interacción y sincronización entre

múltiples dispositivos distribuidos, lo que abriría nuevas oportunidades para la investigación y la optimización de sistemas eléctricos de alta complejidad.

Finalmente, la implementación óptima y el ajuste de estos algoritmos y protocolos no solo mejoran la estabilidad y la precisión de las simulaciones, sino que también optimizan el rendimiento general del sistema. Esto proporciona una base sólida para futuras investigaciones y desarrollos en el campo de la ingeniería eléctrica, con un enfoque en la integración de tecnologías emergentes y la mejora continua de la infraestructura de comunicación y control en los Sistemas Eléctricos de Potencia.

Referencias

- [1] S. Marzal, R. Salas, R. González-Medina, G. Garcerá, y E. Figueres, "Current challenges and future trends in the field of communication architectures for microgrids," *Renewable and Sustainable Energy Reviews*, vol. 82, pp. 3610-3622, 2018.
- [2] W. You y H. Ge, "Design and Implementation of Modbus Protocol for Intelligent Building Security," in 2019 IEEE 19th International Conference on Communication Technology (ICCT), pp. 420-423, 2019. [Online]. Available: <https://doi.org/10.1109/ICCT46805.2019.8946996>.
- [3] S. Mohagheghi, J. Stoupis and Z. Wang, "Communication protocols and networks for power systems-current status and future trends," 2009 IEEE/PES Power Systems Conference and Exposition, Seattle, WA, USA, 2009, pp. 1-9, doi: 10.1109/PSCE.2009.4840174.
- [4] A. Bani-Ahmed, L. Weber, A. Nasiri y H. Hosseini, "Microgrid communications: State of the art and future trends," en 3rd International Conference on Renewable Energy Research and Applications (ICRERA 2014), pp. 780-785, 2014. [En línea]. Disponible en: <https://doi.org/10.1109/ICRERA.2014.7016491>.
- [5] IEEE Standard for Substation Intelligent Electronic Devices (IEDs) Cyber Security Capabilities (IEEE Std C37.240-2014), "IEC 61850-8-1 Communication networks and systems for power utility automation - Part 8-1: Specific communication service mapping (SCSM) - Mappings to MMS (ISO/IEC 9506-1 and ISO/IEC 9506-2) and to ISO/IEC 8802-3".
- [6] R. Schneider, "Modicon Modbus Protocol Reference Guide", Schneider Electric, 2002.
- [7] IEEE Std 1815-2012, "IEEE Standard for Electric Power Systems Communications - Distributed Network Protocol (DNP3)".
- [8] IEC 60870-5-101:2003, "Telecontrol equipment and systems - Part 5-101: Transmission protocols - Companion standard for basic telecontrol task".
- [9] IEC 60870-5-104:2000, "Telecontrol equipment and systems - Part 5-104: Transmission protocols - Network access for IEC 60870-5-101 using standard transport profiles".
- [10] energyx.pe, "Sistemas eléctricos de potencia (SEP): Definición y partes", [en línea]. Disponible en: <https://energyx.pe/sistemas-electricos-de-potencia-sep-definicion-y-partes/> [Accedido el 9 de abril de 2024].
- [11] es.scribd.com, "Protocolos de Comunicación Scada", [en línea]. Disponible en: <https://es.scribd.com/document/450654983/Protocolos-de-Comunicacion-Scada> [Accedido el 9 de abril de 2024].
- [12] Typhoon HIL, "IEC 61850 GOOSE Protocol", en Typhoon HIL Software Manual, disponible en:

https://www.typhoon-hil.com/documentation/typhoon-hil-software-manual/References/iec_61850_goose_protocol.html, [Accedido el 15 de Abril de 2024].

[13] S. Sanati, M. Azzouz, y A. Awad, "An Adaptive Overcurrent Protection for Solar-based DC Microgrids Using IEC 61850," en arXiv preprint arXiv:2307.01940, 2023.

[14] K. Gupta, S. Sahoo, y B. K. Panigrahi, "Delay-aware Semantic Sampling in Power Electronic Systems," IEEE Transactions on Smart Grid, 2023.

[15] X. Zha, C. Yin, J. Sun, M. Huang, and Q. Li, "Improving the Stability and Accuracy of Power Hardware-in-the-Loop Simulation Using Virtual Impedance Method," Energies, vol. 9, no. 11, p. 974, 2016. doi: 10.3390/en9110974.

[16] R. Brandl, "Operational Range of Several Interface Algorithms for Different Power Hardware-In-The-Loop Setups," Energies, vol. 10, no. 12, p. 1946, 2017. doi: [10.3390/en10121946](https://doi.org/10.3390/en10121946).

[17] E. Buraimoh, G. Ozkan, L. Timilsina, P. K. Chamarthi, B. Papari, and C. S. Edrington, "Overview of Interface Algorithms, Interface Signals, Communication and Delay in Real-Time Co-Simulation of Distributed Power Systems," IEEE Access, vol. 11, pp. 103925-103955, 2023. doi: 10.1109/ACCESS.2023.3317250.

[18] A. Aghmadi, H. Hussein, K. H. Polara, and O. Mohammed, "A Comprehensive Review of Architecture, Communication, and Cybersecurity in Networked Microgrid Systems," Inventions, vol. 8, no. 4, p. 84, 2023. doi: [10.3390/inventions8040084](https://doi.org/10.3390/inventions8040084).

[19] M. Ratshitanga, E. F. Orumwense, S. Krishnamurthy, and R. Melamu, "A Review of Demand-Side Resources in Active Distribution Systems: Communication Protocols, Smart Metering, Control, Automation, and Optimization," Appl. Sci., vol. 13, no. 23, p. 12573, 2023. doi: 10.3390/app132312573.

[20] Sistema Eléctrico, TuVeras.com. <https://www.tuveras.com/lineas/sistemaelectrico.htm> (accedido: agosto 19, 2024).

[21] S. Mohagheghi, M. Mousavi, J. Stoupis, and Z. Wang, "Modeling distribution automation system components using IEC 61850," in *Proc. 2009 IEEE Power & Energy Society General Meeting*, Calgary, AB, Canada, 2009, pp. 1-6, doi: 10.1109/PES.2009.5275841.

Se usó ChatGPT para corrección gramatical, traducción y escritura.

Anexos

Se presenta el siguiente código de matlab, el cual sirve para poder obtener los datos obtenidos en el apartado de resultados:

Código Matlab

// Inicialización de variables y parámetros

```
initialize VSS_voltage, PPS_voltage, VSS_current, PPS_current
```

```
initialize Z1, Z2 // Impedancias en VSS y PPS
```

```
initialize Delta_t // Retardo en la interfaz
```

```
initialize time_step, simulation_time
```

```
initialize ITM_transfer_function
```

// Bucle principal de simulación

```
for t = 0 to simulation_time step time_step do
```

// Predicción en el VSS

```
VSS_voltage_pred = VSS_voltage + time_step * f(VSS_voltage, VSS_current)
```

```
VSS_current_pred = VSS_current + time_step * g(VSS_voltage, VSS_current)
```

// Enviar datos predichos a PPS

```
send VSS_voltage_pred to PPS via TCP/IP
```

```
send VSS_current_pred to PPS via TCP/IP
```

// Recibir datos desde PPS

```
receive PPS_voltage_actual from PPS via TCP/IP
```

```
receive PPS_current_actual from PPS via TCP/IP
```

// Aplicar ITM en VSS

```
Z1 = calculate_impedance(VSS)
```

```
Z2 = calculate_impedance(PPS)
```

```
ITM_transfer_function = (Z1/Z2) * exp(s * Delta_t)
```

// Ajustar valores en VSS con el algoritmo ITM

```
VSS_voltage = ITM_transfer_function * PPS_voltage_actual
```

```
VSS_current = ITM_transfer_function * PPS_current_actual
```

// Predicción en el PPS

```
PPS_voltage_pred = PPS_voltage + time_step * h(PPS_voltage, PPS_current)
```

```
PPS_current_pred = PPS_current + time_step * k(PPS_voltage, PPS_current)
```

// Enviar datos predichos a VSS

```
send PPS_voltage_pred to VSS via TCP/IP
```

```
send PPS_current_pred to VSS via TCP/IP
```

// Recibir datos desde VSS

```

receive VSS_voltage_actual from VSS via TCP/IP
receive VSS_current_actual from VSS via TCP/IP
// Aplicar ITM en PPS
Z1 = calculate_impedance(VSS)
Z2 = calculate_impedance(PPS)
ITM_transfer_function = (Z1/Z2) * exp(s * Delta_t)
// Ajustar valores en PPS con el algoritmo ITM
PPS_voltage = ITM_transfer_function * VSS_voltage_actual
PPS_current = ITM_transfer_function * VSS_current_actual
// Calcular otras variables relevantes
VSS_RMS_voltage = sqrt(mean(VSS_voltage^2))
VSS_RMS_current = sqrt(mean(VSS_current^2))
PPS_RMS_voltage = sqrt(mean(PPS_voltage^2))
PPS_RMS_current = sqrt(mean(PPS_current^2))
power_P = VSS_RMS_voltage * VSS_RMS_current * cos(phase_angle(VSS_voltage, VSS_current))
power_Q = VSS_RMS_voltage * VSS_RMS_current * sin(phase_angle(VSS_voltage, VSS_current))
// Guardar datos para análisis posterior
store VSS_voltage, VSS_current, PPS_voltage, PPS_current, power_P, power_Q

end for

```

Pseudocódigo AITM

```

// Pseudocódigo para el Algoritmo AITM (Advanced Ideal Transformer Model)

// Inicializar parámetros
Definir Z1(s)      // Impedancia del subsistema simulado
Definir Z2(s)      // Impedancia del hardware o dispositivo bajo prueba
Definir ZC(s)      // Impedancia de compensación para mejorar la estabilidad
Definir Δt         // Retardo de la interfaz (incluye tiempos de procesamiento y amplificación)
Definir G0(s)      // Función de transferencia del sistema

// Calcular la impedancia equivalente del sistema AITM
 $Z_{eq}(s) = Z_C(s) + Z_2(s)$ 

// Calcular la función de transferencia del AITM
 $G_0(s) = (Z_1(s) / Z_{eq}(s)) * e^{(-s\Delta t)}$ 

```

```

// Iterar sobre el tiempo de simulación
Para cada paso de tiempo k en la simulación:
    // Obtener señales de entrada (voltaje y corriente)
    Obtener V_input(k) // Voltaje de entrada en el subsistema 1
    Obtener I_input(k) // Corriente de entrada en el subsistema 2

    // Aplicar la función de transferencia G0(s) para calcular la salida
    V_output(k) = G0(s) * V_input(k)
    I_output(k) = G0(s) * I_input(k)

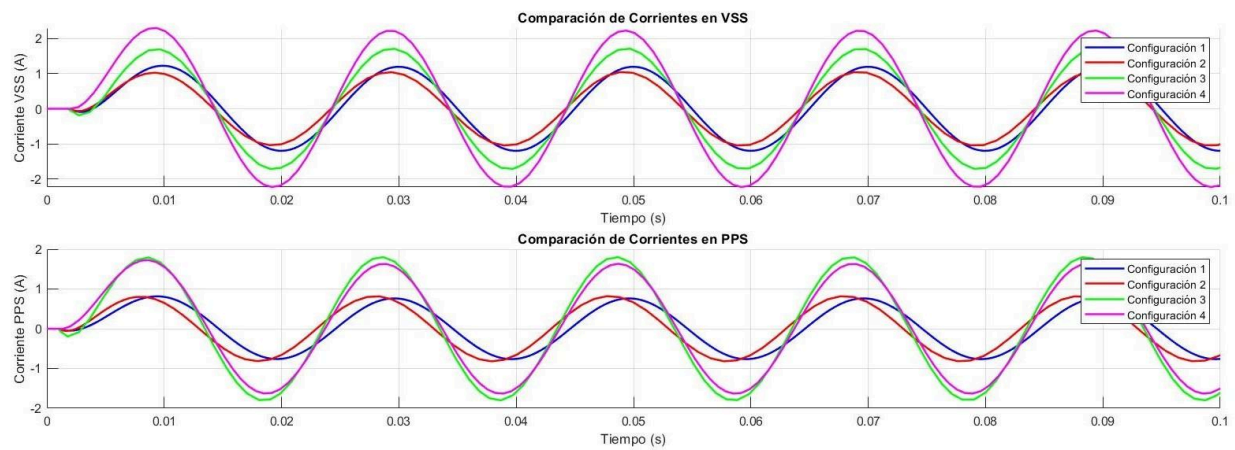
    // Actualizar las señales en el subsistema 2 con la salida del subsistema 1
    V_PPS(k) = V_output(k) // Actualizar el voltaje en el subsistema 2 (PPS)
    I_PPS(k) = I_output(k) // Actualizar la corriente en el subsistema 1 (VSS)

    // Monitorear la estabilidad del sistema
    Si  $|Z1(s) / Z_{eq}(s)| > \text{Límite de estabilidad}$ :
        // Aplicar corrección en ZC(s)
        Ajustar ZC(s) para mejorar la estabilidad

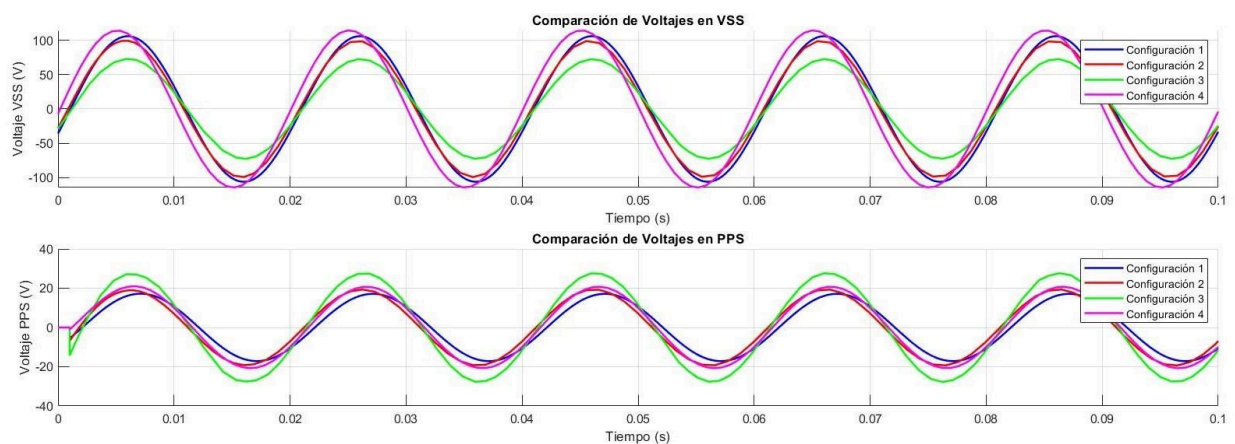
    // Guardar resultados para análisis posterior
    Guardar V_PPS(k), I_PPS(k) // Guardar voltaje y corriente en PPS
    Guardar V_output(k), I_output(k) // Guardar señales de salida para análisis

// Fin de la simulación

```

Anexo 1 AITM Comparación de Corrientes entre el lado VSS y el PPS.



Anexo 2 AITM Comparación de Voltajes entre el lado VSS y el PPS.

Pseudocódigo PCD

// Pseudocódigo para el Algoritmo PCD (Partial Circuit Duplication)

// Inicializar parámetros

Definir $Z1(s)$ // Impedancia del subsistema 1 (VSS)

Definir $Z2(s)$ // Impedancia del subsistema 2 (PPS)

Definir $Z12(s)$ // Impedancia de acoplamiento adicional entre $Z1$ y $Z2$

Definir Δt // Retardo de la interfaz (incluye tiempos de procesamiento y amplificación)

Definir $G0(s)$ // Función de transferencia del sistema

// Calcular la función de transferencia del PCD

$G0(s) = (Z1(s) * Z2(s)) / ((Z1(s) + Z12(s)) * (Z2(s) + Z12(s))) * e^{(s\Delta t)}$

// Iterar sobre el tiempo de simulación

Para cada paso de tiempo k en la simulación:

 // Obtener señales de entrada (voltaje y corriente)

 Obtener $V_input(k)$ // Voltaje de entrada en el subsistema 1

 Obtener $I_input(k)$ // Corriente de entrada en el subsistema 2

 // Aplicar la función de transferencia $G0(s)$ para calcular la salida

$V_output(k) = G0(s) * V_input(k)$

$I_output(k) = G0(s) * I_input(k)$

 // Actualizar las señales en los subsistemas 1 y 2

$V_PPS(k) = V_output(k)$ // Actualizar el voltaje en el subsistema 2 (PPS)

$I_VSS(k) = I_output(k)$ // Actualizar la corriente en el subsistema 1 (VSS)

// Iteración para lograr la solución precisa

Para cada iteración i hasta que se alcance la convergencia:

 // Calcular nuevo $V_output(k)$ e $I_output(k)$ basado en las iteraciones anteriores

$V_output(k) = (V_input(k) * Z2(s)) / (Z2(s) + Z12(s)) + (I_input(k) * Z12(s))$

$I_output(k) = (I_input(k) * Z1(s)) / (Z1(s) + Z12(s)) + (V_input(k) * Z12(s))$

 // Actualizar los valores en los subsistemas

$V_PPS(k) = V_output(k)$

$I_VSS(k) = I_output(k)$

```

// Verificar convergencia
    Si  $|V_{\text{output}}(k) - V_{\text{output\_prev}}(k)| < \text{Tolerancia}$  y  $|I_{\text{output}}(k) - I_{\text{output\_prev}}(k)| < \text{Tolerancia}$ :
        Romper el bucle de iteración
    Fin si

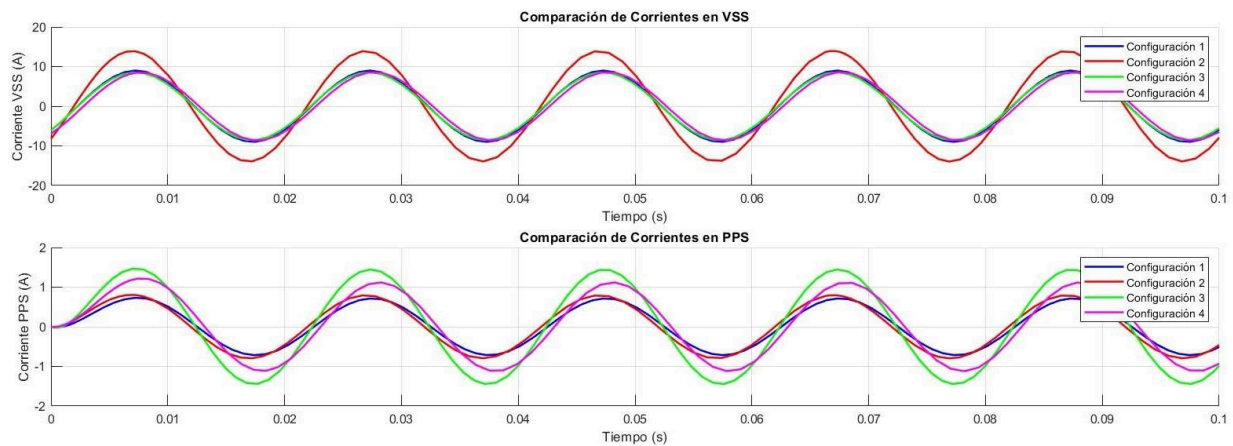
// Guardar los valores actuales como anteriores para la próxima iteración
 $V_{\text{output\_prev}}(k) = V_{\text{output}}(k)$ 
 $I_{\text{output\_prev}}(k) = I_{\text{output}}(k)$ 

Fin de la iteración

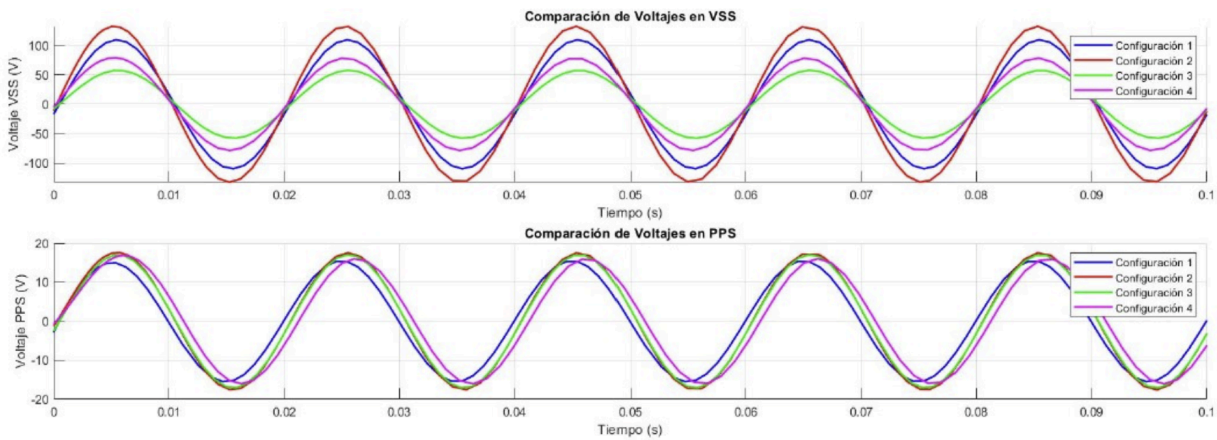
// Guardar resultados para análisis posterior
Guardar  $V_{\text{PPS}}(k)$ ,  $I_{\text{VSS}}(k)$  // Guardar voltaje y corriente en PPS y VSS
Guardar  $V_{\text{output}}(k)$ ,  $I_{\text{output}}(k)$  // Guardar señales de salida para análisis

// Fin de la simulación

```



Anexo 3 PCD Comparación de Corrientes entre el lado VSS y el PPS.



Anexo 4 PCD Comparación de Voltajes entre el lado VSS y el PPS.

Pseudocódigo DIM

// Pseudocódigo para el Algoritmo DIM (Damping Impedance Method)

// Inicializar parámetros

Definir Z1(s) // Impedancia del subsistema 1 (VSS)

Definir Z2(s) // Impedancia del subsistema 2 (PPS)

Definir Z12(s) // Impedancia de acoplamiento adicional entre Z1 y Z2

Definir Z*(s) // Impedancia de amortiguamiento

Definir Δt // Retardo de la interfaz (incluye tiempos de procesamiento y amplificación)

Definir G0(s) // Función de transferencia del sistema

// Calcular la función de transferencia del DIM

$$G0(s) = (Z1(s) * (Z2(s) - Z^*(s))) / ((Z1(s) + Z12(s)) * (Z2(s) + Z12(s) + Z^*(s))) * e^{(-s\Delta t)}$$

// Iterar sobre el tiempo de simulación

Para cada paso de tiempo k en la simulación:

// Obtener señales de entrada (voltaje y corriente)

Obtener V_input(k) // Voltaje de entrada en el subsistema 1

Obtener I_input(k) // Corriente de entrada en el subsistema 2

// Aplicar la función de transferencia G0(s) para calcular la salida

$$V_output(k) = G0(s) * V_input(k)$$

$$I_output(k) = G0(s) * I_input(k)$$

// Actualizar las señales en los subsistemas 1 y 2

V_PPS(k) = V_output(k) // Actualizar el voltaje en el subsistema 2 (PPS)

I_VSS(k) = I_output(k) // Actualizar la corriente en el subsistema 1 (VSS)

// Ajustar la impedancia de amortiguamiento $Z^*(s)$

Si el sistema se vuelve inestable o se detectan oscilaciones:

// Ajustar $Z^*(s)$ para mejorar la estabilidad

Si $Z^*(s) < Z_2(s)$:

Incrementar $Z^*(s)$ ligeramente

Sino:

Reducir $Z^*(s)$ ligeramente

Fin si

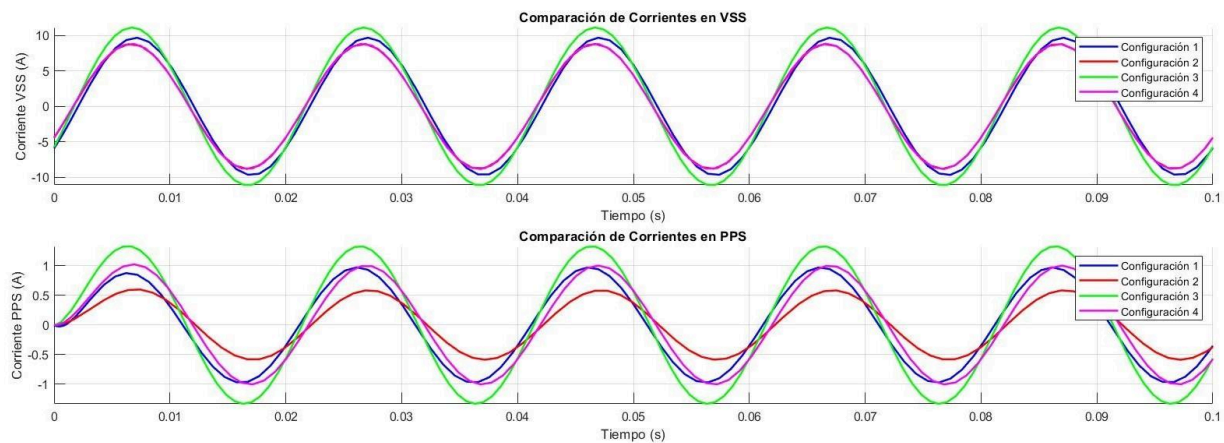
Fin si

// Guardar resultados para análisis posterior

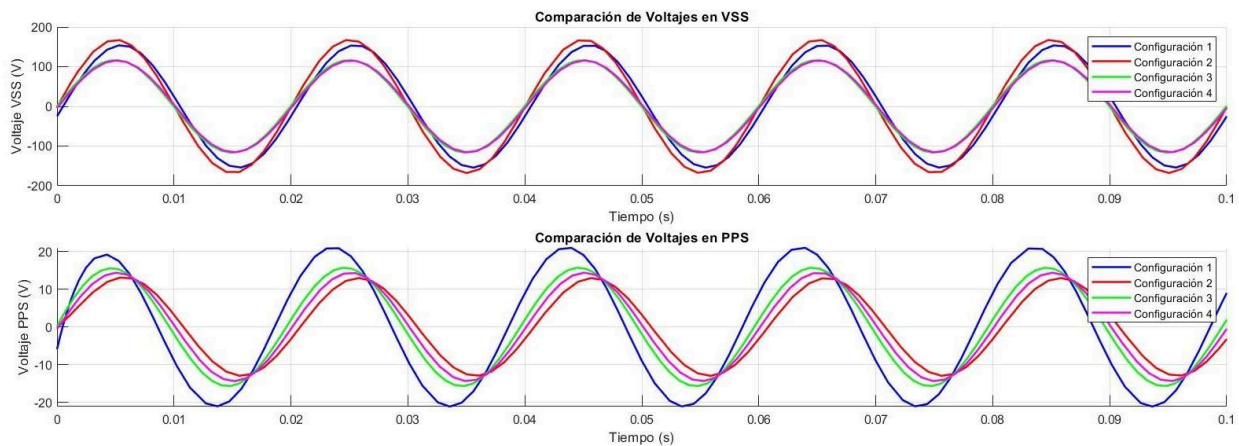
Guardar V_PPS(k), I_VSS(k) // Guardar voltaje y corriente en PPS y VSS

Guardar V_output(k), I_output(k) // Guardar señales de salida para análisis

// Fin de la simulación



Anexo 5 DIM Comparación de Corrientes entre el lado VSS y el PPS.



Anexo 6 DIM Comparación de Voltajes entre el lado VSS y el PPS.

Pseudocódigo FCF/ITM

// Pseudocódigo para el Algoritmo FCF (Feedback Current Filter)

// Inicializar parámetros

Definir $Z1(s)$ // Impedancia del subsistema 1 (VSS)

Definir $Z2(s)$ // Impedancia del subsistema 2 (PPS)

Definir Δt // Retardo de la interfaz (incluye tiempos de procesamiento y amplificación)

Definir $TFCF(s)$ // Función de transferencia del filtro de corriente de retroalimentación

Definir $G0(s)$ // Función de transferencia del sistema

// Configurar el filtro de retroalimentación

Configurar filtro FCF como un filtro pasa-bajo con frecuencia de corte f_c

// Calcular la función de transferencia del FCF

$$G0(s) = TFCF(s) * (Z1(s) / Z2(s)) * e^{(s\Delta t)}$$

// Iterar sobre el tiempo de simulación

Para cada paso de tiempo k en la simulación:

 // Obtener señales de entrada (voltaje y corriente)

 Obtener $V_{input}(k)$ // Voltaje de entrada en el subsistema 1

 Obtener $I_{input}(k)$ // Corriente de entrada en el subsistema 2

```

// Aplicar la función de transferencia G0(s) para calcular la salida
V_output(k) = G0(s) * V_input(k)
I_output(k) = G0(s) * I_input(k)

// Aplicar el filtro FCF a las señales de salida
I_output_filtered(k) = Filtro_PasaBajo(I_output(k), f_c)
V_output_filtered(k) = Filtro_PasaBajo(V_output(k), f_c)

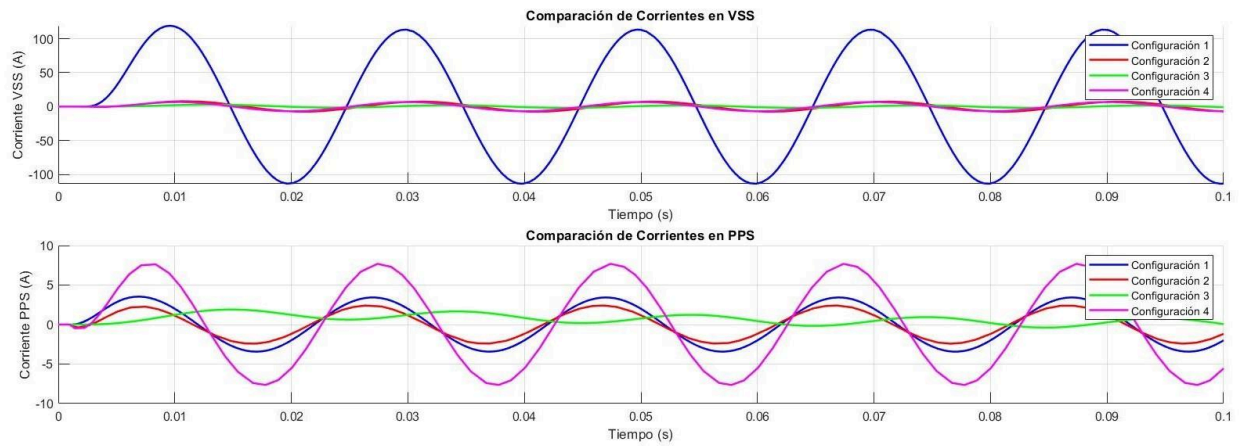
// Actualizar las señales en los subsistemas 1 y 2
V_PPS(k) = V_output_filtered(k) // Actualizar el voltaje en el subsistema 2 (PPS)
I_VSS(k) = I_output_filtered(k) // Actualizar la corriente en el subsistema 1 (VSS)

// Monitorear el rendimiento del sistema
Si se detectan oscilaciones o inestabilidad en las señales:
    // Ajustar los parámetros del filtro FCF
    Si la frecuencia de corte f_c es demasiado baja:
        Incrementar f_c ligeramente
    Sino si f_c es demasiado alta:
        Reducir f_c ligeramente
    Fin si
Fin si

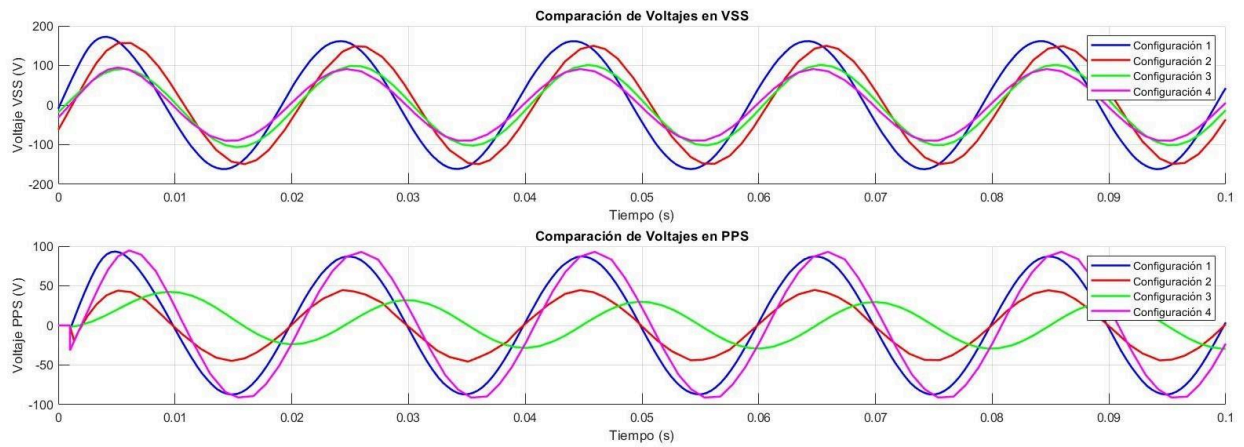
// Guardar resultados para análisis posterior
Guardar V_PPS(k), I_VSS(k) // Guardar voltaje y corriente en PPS y VSS
Guardar V_output_filtered(k), I_output_filtered(k) // Guardar señales filtradas para análisis

// Fin de la simulación

```



Anexo 7 FCF/ITM Comparación de Corrientes entre el lado VSS y el PPS.



Anexo 8 FCF/ITM Comparación de Voltajes entre el lado VSS y el PPS.

Pseudocódigo TLM

// Pseudocódigo para el Algoritmo TLM (Transmission Line Model)

// Inicializar parámetros

Definir L // Inductancia en el enlace de transmisión

Definir C // Capacitancia en el enlace de transmisión

Definir Δt // Paso de tiempo de la simulación

Definir R_{t_L} // Impedancia equivalente para inductancia ($R_t = L / \Delta t$)

Definir R_{t_C} // Impedancia equivalente para capacitancia ($R_t = \Delta t / C$)

Definir $Z_1(s)$ // Impedancia del subsistema 1 (VSS)

Definir $Z_2(s)$ // Impedancia del subsistema 2 (PPS)

Definir $G_0(s)$ // Función de transferencia del sistema

// Calcular las impedancias equivalentes

$R_{t_L} = L / \Delta t$

$R_{t_C} = \Delta t / C$

// Calcular el coeficiente α

$\alpha = (Z_L - R_{t_L}) / (Z_L + R_{t_L})$ // Z_L es la impedancia de carga (puede ser Z_1 o Z_2 según el caso)

// Calcular la función de transferencia del TLM

$G_0(s) = [(1 - \alpha * e^{(-2s\Delta t)}) / (1 + \alpha * e^{(-2s\Delta t)})] * (Z_1(s) / R_{t_L})$

// Iterar sobre el tiempo de simulación

Para cada paso de tiempo k en la simulación:

 // Obtener señales de entrada (voltaje y corriente)

 Obtener $V_{input}(k)$ // Voltaje de entrada en el subsistema 1

 Obtener $I_{input}(k)$ // Corriente de entrada en el subsistema 2

 // Aplicar la función de transferencia $G_0(s)$ para calcular la salida en el subsistema 2 (PPS)

$V_{output_PPS}(k) = G_0(s) * V_{input}(k)$

$I_{output_PPS}(k) = G_0(s) * I_{input}(k)$

 // Aplicar la función de transferencia $G_0(s)$ para calcular la salida en el subsistema 1 (VSS)

$V_{output_VSS}(k) = G_0(s) * V_{input}(k)$

$I_{output_VSS}(k) = G_0(s) * I_{input}(k)$

```

// Actualizar las señales en los subsistemas 1 y 2
V_PPS(k) = V_output_PPS(k) // Actualizar el voltaje en el subsistema 2 (PPS)
I_VSS(k) = I_output_VSS(k) // Actualizar la corriente en el subsistema 1 (VSS)

// Monitorear la estabilidad del sistema
Si el sistema muestra signos de inestabilidad:
    // Ajustar los parámetros de la simulación o verificar la convergencia de los valores
    Ajustar  $\alpha$  o  $\Delta t$  si es necesario para mejorar la estabilidad
Fin si

// Guardar resultados para análisis posterior
Guardar V_PPS(k), I_VSS(k) // Guardar voltaje y corriente en PPS y VSS
Guardar V_output_PPS(k), I_output_PPS(k) // Guardar señales de salida del PPS para análisis
Guardar V_output_VSS(k), I_output_VSS(k) // Guardar señales de salida del VSS para análisis

// Fin de la simulación

```

Pseudocódigo TFA

```

// Pseudocódigo para el Algoritmo TFA (Time-Variant First-Order Approximation)

// Inicializar parámetros
Definir a, b // Coeficientes del sistema (dependerán del tipo de HUT: inductivo/resistivo o
capacitivo/resistivo)
Definir  $\Delta t$  // Paso de tiempo de la simulación
Definir  $i_2(k)$ ,  $i_2(k-1)$ ,  $i_2(k-2)$  // Corrientes en el HUT en los pasos de tiempo actuales y
anteriores
Definir  $v_1(k)$ ,  $v_1(k-1)$ ,  $v_1(k-2)$  // Voltajes en el HUT en los pasos de tiempo actuales y anteriores
Definir  $R_{eq}(k)$ ,  $I_{eq}(k)$  // Resistencia y corriente equivalentes en cada paso de tiempo
Definir  $\alpha$ ,  $\beta$  // Factores de actualización en cada paso de tiempo
Definir  $G_{eq}(k)$  // Conductancia equivalente

// Inicialización de condiciones iniciales
Inicializar  $i_2(0)$ ,  $v_1(0)$ ,  $R_{eq}(0)$ ,  $I_{eq}(0)$ 
Inicializar  $\alpha$ ,  $\beta$  en función de a, b, y  $\Delta t$ 

// Iterar sobre el tiempo de simulación

```

Para cada paso de tiempo k en la simulación:

// Actualizar los coeficientes α y β

$$\alpha = (2 - a * \Delta t) / (2 + a * \Delta t)$$

$$\beta = (2 * b * \Delta t) / (2 + a * \Delta t)$$

// Calcular la corriente y resistencia equivalentes

$$Req(k) = \alpha * v1(k-1) + \beta * i2(k-1)$$

$$leq(k) = \beta * i2(k-1)$$

// Calcular la corriente en el HUT usando la aproximación TFA

$$i2(k) = Req(k) * v1(k) + leq(k)$$

// Guardar los valores para el siguiente paso

$$i2(k-2) = i2(k-1)$$

$$i2(k-1) = i2(k)$$

$$v1(k-2) = v1(k-1)$$

$$v1(k-1) = v1(k)$$

// Verificar la estabilidad numérica

Si la matriz de coeficientes (basada en α y β) se vuelve singular:

// Conservar las estimaciones anteriores de $Req(k)$ y $leq(k)$

$$Req(k) = Req(k-1)$$

$$leq(k) = leq(k-1)$$

// Calcular la fuente de corriente de compensación $Ic(k)$

$$e(k) = i2(k) - i1(k)$$

$$Ic(k+1) = Ic(k) + KI * e(k)$$

// Ajustar la corriente $i2(k)$ con la compensación

$$i2(k) = i2(k) + Ic(k+1)$$

Fin si

// Guardar resultados para análisis posterior

Guardar $i2(k)$, $v1(k)$ // Guardar corriente y voltaje para análisis

Guardar $Req(k)$, $leq(k)$ // Guardar resistencia y corriente equivalentes

// Fin de la simulación

Tabla 3 Comparación del algoritmo de AIMT con diferentes configuraciones.

Configuración	Voltaje (V)	Corriente (A)	Potencia (W)
Configuración 1	Media: -0.4400 Desviación: 28.2550 Máxima: 40.2885 Mínima: -40.3385	Media: 0.0123 Desviación: 0.3136 Máxima: 0.4572 Mínima: -0.4586	Media: 27.9076 Desviación: 7.5232 Máxima: 31.5740 Mínima: -0.2471
Configuración 2	Media: 0.2912 Desviación: 9.4550 Máxima: 38.0993 Mínima: -27.7491	Media: 0.0157 Desviación: 0.2834 Máxima: 0.4168 Mínima: -0.4170	Media: 10.3234 Desviación: 3.4977 Máxima: 12.0625 Mínima: -3.0050
Configuración 3	Media: 0.5716 Desviación: 47.1446 Máxima: 66.8457 Mínima: -66.8063	Media: 0.0112 Desviación: 0.3445 Máxima: 0.4957 Mínima: -0.4958	Media: -55.6068 Desviación: 13.4692 Máxima: 0.0000 Mínima: -62.9364
Configuración 4	Media: -0.7222 Desviación: 35.8217 Máxima: 51.5463 Mínima: -51.7019	Media: 0.0223 Desviación: 0.4791 Máxima: 0.6995 Mínima: -0.7000	Media: 37.8423 Desviación: 10.2272 Máxima: 43.0743 Mínima: -1.0888

Tabla 4 Comparación del algoritmo de TLM con diferentes configuraciones.

Configuración	Voltaje (V)	Corriente (A)	Potencia (W)
Configuración Única	Media: -0.3721 Desviación: 0.7986 Máxima: 0.7956 Mínima: -1.5293	Media: -0.0740 Desviación: 0.0596 Máxima: 0.0269 Mínima: -0.1634	Media: 0.0323 Desviación: 0.0099 Máxima: 0.0397 Mínima: 0.0005

Tabla 5 Comparación del algoritmo de PCD con diferentes configuraciones.

Configuración	Voltaje (V)	Corriente (A)	Potencia (W)
Configuración 1	Media: -59.7974 Desviación: 79.0804 Máxima: 49.9669 Mínima: -173.3230	Media: -5.1823 Desviación: 6.2985 Máxima: 4.0637 Mínima: -14.4394	Media: 136.4648 Desviación: 34.0323 Máxima: 265.1924 Mínima: 27.9641
Configuración 2	Media: -108.4055 Desviación: 96.1999 Máxima: 37.4983 Mínima: -243.7834	Media: -6.3668 Desviación: 9.9581 Máxima: 8.0442 Mínima: -20.6345	Media: 158.7366 Desviación: 14.7174 Máxima: 168.0780 Mínima: 44.2506
Configuración 3	Media: -46.5966 Desviación: 41.8575 Máxima: 15.5376 Mínima: -105.8271	Media: -4.7964 Desviación: 6.1362 Máxima: 4.2737 Mínima: -13.6210	Media: 1.3314 Desviación: 7.7748 Máxima: 50.8126 Mínima: -2.0993
Configuración 4	Media: -44.2642 Desviación: 56.5983 Máxima: 41.2201 Mínima: -123.5860	Media: -6.0708 Desviación: 6.1840 Máxima: 2.9349 Mínima: -14.9443	Media: 42.7301 Desviación: 14.7526 Máxima: 111.7944 Mínima: 1.0782

Tabla 6 Comparación del algoritmo de DIM con diferentes configuraciones.

Configuración	Voltaje (V)	Corriente (A)	Potencia (W)
Configuración 1	Media: 0.2553 Desviación: 97.2872 Máxima: 137.8965 Mínima: -137.4765	Media: 0.0025 Desviación: 6.1965 Máxima: 8.8275 Mínima: -8.7491	Media: 49.2974 Desviación: 37.0228 Máxima: 208.3878 Mínima: -18.5644
Configuración 2	Media: -0.8262 Desviación: 28.0831 Máxima: 40.5769 Mínima: -40.8214	Media: -0.0123 Desviación: 5.8243 Máxima: 8.2376 Mínima: -8.2379	Media: 46.0595 Desviación: 17.3714 Máxima: 106.5410 Mínima: -4.3698
Configuración 3	Media: 0.2595 Desviación: 11.6316 Máxima: 16.8274 Mínima: -16.7955	Media: -0.0108 Desviación: 6.9275 Máxima: 9.7926 Mínima: -9.7926	Media: 54.6957 Desviación: 14.5841 Máxima: 86.1907 Mínima: -22.7302
Configuración 4	Media: 0.1103 Desviación: 7.2786 Máxima: 10.4418 Mínima: -10.2710	Media: -0.0114 Desviación: 5.5915 Máxima: 7.9063 Mínima: -7.9063	Media: 151.3760 Desviación: 23.2952 Máxima: 170.6916 Mínima: 11.1141

Configuración	Voltaje (V)	Corriente (A)	Potencia (W)
Configuración 1	Media: 0.2631	Media: 0.3694	Media: 3.0191
	Desviación: 33.8345	Desviación: 4.9296	Desviación: 61.7584
	Máxima: 65.0445	Máxima: 7.5370	Máxima: 93.0074
	Mínima: -47.3067	Mínima: -7.1421	Mínima: -267.5757
Configuración 2	Media: 2.5614	Media: 0.4116	Media: 2.4345
	Desviación: 31.9389	Desviación: 5.1378	Desviación: 55.9632
	Máxima: 95.2763	Máxima: 8.0331	Máxima: 96.2046
	Mínima: -62.3757	Mínima: -7.5498	Mínima: -283.2141
Configuración 3	Media: -10.7006	Media: -0.3600	Media: 11.1700
	Desviación: 81.8196	Desviación: 0.9986	Desviación: 9.3653
	Máxima: 117.6656	Máxima: 1.1882	Máxima: 28.1446
	Mínima: -148.2190	Mínima: -2.1631	Mínima: -5.1802
Configuración 4	Media: -0.0879	Media: 0.1738	Media: -43.2078
	Desviación: 20.5246	Desviación: 3.8643	Desviación: 24.0810
	Máxima: 32.8326	Máxima: 5.7162	Máxima: 0.0000
	Mínima: -31.4532	Mínima: -5.6343	Mínima: -151.3832

[illegible]