

Trabajo Fin de Máster

Máster en Sistemas de Energía Eléctrica

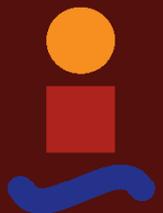
Cosimulación de redes de distribución  
utilizando la plataforma HELICS

Autor: Elvis Junior Mármol Carrión

Tutor: Alejandro Marano Marcolini

Dpto. Ingeniería Eléctrica  
Escuela Técnica Superior de Ingeniería  
Universidad de Sevilla

Sevilla, 2024





Trabajo Fin de Máster  
Máster en Sistemas de Energía Eléctrica

# **Cosimulación de redes de distribución utilizando la plataforma HELICS**

Autor:

Elvis Junior Mármol Carrión

Tutor:

Alejandro Marano Marcolini

Profesor titular

Dpto. Ingeniería Eléctrica  
Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2024



Trabajo Fin de Máster: Cosimulación de redes de distribución utilizando la plataforma HELICS

Autor: Elvis Junior Mármol Carrión

Tutor: Alejandro Marano Marcolini

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

acuerdan otorgarle la calificación de:

El Secretario del Tribunal

Fecha:



*A mi familia.*

*A mi Dios.*



# Agradecimientos

---

**E**n primer lugar, quiero agradecer a Dios, porque gracias a Él todo esto ha sido posible, como lo indica Su palabra. Dios, gracias por dotarme de las herramientas y aptitudes necesarias para seguir adelante en cada momento hasta completar esta gran etapa.

Agradezco a mis padres, Colasa Carrión y Fred Mármol, mi mayor motivación en la vida y modelos a seguir. Gracias por brindarme su apoyo incondicional a pesar de la distancia; nada de esto hubiera sido posible sin ustedes. También a mis hermanos, quienes me motivan a ser mejor cada día. De forma especial, quiero agradecer a mi novia, Paola Camilo, quien siempre estuvo a mi lado durante todo este trayecto. Te agradezco por comprenderme a pesar de la distancia.

A mis compañeros de piso, en especial a Osvaldo Villaman y Robert Ramos, así como a mis compañeros de máster, gracias por todo el apoyo y por siempre motivarme a ser exigente conmigo mismo. Asimismo, le agradezco al Ing. Andrés Simó por su plena disposición y soporte incondicional durante este proceso.

A todos los profesionales que han sido maestros de la vida y han dejado su huella en mi crecimiento profesional, especialmente a mi tutor, Alejandro Marano Marcolini, gracias por su comprensión y su plena disposición. Igualmente, expreso mi agradecimiento al Departamento de Ingeniería Eléctrica de la Universidad de Sevilla por brindarme la oportunidad de enriquecer mis conocimientos en este destacado centro académico.

Finalmente, a todas las personas que de una forma directa o indirecta aportaron su granito de arena para hacer posible alcanzar esta meta: ¡A todos, gracias!

*Elvis Junior Mármol Carrión*

*Sevilla, 2024*



# Resumen

---

En el apasionante ámbito de los sistemas de energía eléctrica, la optimización y análisis de los sistemas de distribución son cruciales para garantizar un suministro confiable y eficiente. Una solución innovadora que ha surgido en este contexto es la 'co-simulación', que permite la combinación de diferentes simuladores para obtener análisis más avanzados y precisos en sistemas de energía eléctrica complejos. En lugar de depender de un solo programa para comprender todo el sistema, se fragmenta el problema en partes más pequeñas que se analizan con programas especializados. Posteriormente, se conectan y se establece una comunicación en tiempo real entre estos programas, recreando así la totalidad del sistema.

Hoy en día, la co-simulación ha captado la atención de la comunidad de investigadores, no solo en el ámbito de los sistemas de energía eléctrica, sino en múltiples disciplinas de la ingeniería. Esto se debe a su potencial para construir plataformas digitales que modelan y operan sistemas de ingeniería más complejos.

A raíz de eso, en el presente trabajo se realiza una revisión exhaustiva del estado del arte de la co-simulación en el ámbito de los sistemas eléctricos. El análisis de la literatura se inicia con una presentación de los fundamentos teóricos relacionados con la co-simulación, este abarca múltiples aspectos y características relevantes sobre esta técnica. Se engloban todos los conceptos y elementos que componen esta técnica, para luego analizar las principales herramientas y/o plataformas de co-simulación existentes, destacando especialmente HELICS (Hierarchical Engine for Large-scale Infrastructure Co-Simulation).

También se proporciona un resumen de los resultados obtenidos en diversos casos de estudio y/o prácticos donde HELICS ha sido implementado con éxito como una herramienta y/o plataforma de co-simulación. Finalmente, se presenta las conclusiones derivadas de este estudio.



# Abstract

---

In the exciting field of electrical power systems, optimization and analysis of distribution systems are crucial to ensure reliable and efficient supply. An innovative solution that has emerged in this context is 'co-simulation', which allows the combination of different simulators for more advanced and accurate analysis of complex power systems. Instead of relying on a single program to understand the entire system, the problem is broken down into smaller parts that are analyzed with specialized programs. They are then connected and real-time communication is established between these programs, thus recreating the entire system.

Today, co-simulation has captured the attention of the research community, not only in the field of electric power systems, but in multiple engineering disciplines. This is due to its potential to build digital platforms that model and operate more complex engineering systems.

As a result, a comprehensive review of the state of the art of co-simulation in the field of power systems is carried out in this paper. The literature review begins with a presentation of the theoretical foundations related to co-simulation, which covers multiple aspects and characteristics relevant to this technique. All the concepts and elements that compose this technique are encompassed, to then analyze the main existing co-simulation tools and/or platforms, highlighting especially HELICS (Hierarchical Engine for Large-scale Infrastructure Co-Simulation).

A summary of the results obtained in several case studies and/or practical cases where HELICS has been successfully implemented as a co-simulation tool and/or platform is also provided. Finally, the conclusions derived from this study are presented.

- Translated with "DeepL.com" -



# Índice

---

<b>Agradecimientos</b>	<b>i</b>
<b>Resumen</b>	<b>iii</b>
<b>Abstract</b>	<b>v</b>
<b>Índice</b>	<b>vii</b>
<b>Índice de Tablas</b>	<b>ix</b>
<b>Índice de Figuras</b>	<b>xi</b>
<b>1 Generalidades</b>	<b>1</b>
1.1 <i>Introducción</i>	1
1.2 <i>Objetivos</i>	2
1.2.1 <i>Objetivo General</i>	2
1.2.2 <i>Objetivos Específicos</i>	2
1.3 <i>Metodología</i>	2
1.4 <i>Contenido</i>	3
<b>2 Co-simulación: Estado del Arte</b>	<b>5</b>
2.1 <i>Concepto de 'Co-simulación'</i>	5
2.2 <i>Partes de una Co-simulación</i>	8
2.3 <i>Simuladores</i>	8
2.4 <i>Modelos de Simulación</i>	9
2.4.1 <i>Modelos de Simulación según su Temporalidad</i>	9
2.4.2 <i>Modelos de Simulación según su Incertidumbre</i>	10
2.4.3 <i>Modelos de Simulación según Naturaleza del Cambio</i>	11
2.5 <i>Infraestructura de Tiempo de Ejecución</i>	11
2.5.1 <i>Infraestructura de Simulación Paralela</i>	11
2.5.2 <i>Infraestructura de Simulación Distribuida</i>	12
2.6 <i>Interfaz de Sincronización</i>	12
2.7 <i>Tipos de simulaciones</i>	15
2.7.1 <i>Simulaciones de Eventos Discretos</i>	15
2.7.2 <i>Simulaciones de Tiempo Continuo</i>	16
2.7.3 <i>Simulaciones Híbridas</i>	16
2.8 <i>Estándares de Co-simulación</i>	17
2.8.1 <i>High Level Architecture (HLA)</i>	17
2.8.2 <i>Functional Mockup Interface (FMI)</i>	18
2.8.3 <i>Ad-hoc Architectural Style</i>	18
2.9 <i>Etapas Generales de la Co-simulación</i>	19
2.9.1 <i>Etapa 1: Creación</i>	19
2.9.2 <i>Etapa 2: Inicialización</i>	19
2.9.3 <i>Etapa 3: Ejecución</i>	21
2.9.4 <i>Etapa 4: Estado Final</i>	21
2.10 <i>Retos y Limitaciones de la Co-simulación</i>	22
2.10.1 <i>Latencia en la Comunicación e Intercambio de Datos</i>	22
2.10.2 <i>Problemas de Escalabilidad en Co-simulaciones a Gran Escala</i>	22
2.10.3 <i>Validación y Verificación de Modelos de Co-simulación</i>	23

2.10.4	Otros Desafíos de la Co-simulación	23
2.11	<i>Importancia de la Co-simulación</i>	24
2.11.1	Aplicaciones de la Co-simulación en la Actualidad	26
2.11.2	La Co-simulación como Herramienta de Colaboración Interdisciplinaria	27
2.11.3	La Co-simulación en el ámbito de las Redes de Distribución	27
<b>3</b>	<b>Entornos de Co-Simulación con Enfoque en Sistemas Eléctricos</b>	<b>29</b>
3.1	<i>Estado Actual de los Entornos de Co-simulación</i>	29
3.1.1	Principales áreas de Investigación de los Entornos de Co-simulación	29
3.2	<i>Principales Plataformas de Co-simulación Eléctrica</i>	31
3.2.1	CyDER	31
3.2.2	DACCOSIM-NG	32
3.2.3	EPOCHS	33
3.2.4	FNCS	34
3.2.5	GECO	35
3.2.6	HELICS	35
3.2.7	INSPIRE	36
3.2.8	MECSYCO	37
3.2.9	MOSAIK	37
3.2.10	VPNET	38
3.3	<i>Simuladores más utilizados en Co-simulaciones</i>	40
<b>4</b>	<b>Casos de Estudio mediante HELICS</b>	<b>43</b>
4.1	<i>Caso de Estudio 1. Redes de Transmisión-Distribución Multitemporal Escalable (SMTD)</i>	43
4.2	<i>Caso de Estudio 2. Co-simulación para Análisis de Redes Eléctricas y de Gas Natural</i>	44
4.3	<i>Caso de Estudio 3. Co-simulación para el Análisis del Control de Voltaje en Redes Eléctricas</i>	47
4.4	<i>Caso de Estudio 4. Demanda Sensible al Precio en Interacciones TSO-DSO</i>	48
4.5	<i>Caso de Estudio 5. Evaluación de la Respuesta en Frecuencia en Redes de T&amp;D</i>	49
<b>5</b>	<b>Conclusiones</b>	<b>53</b>
	<b>Referencias</b>	<b>55</b>
	<b>Anexo A. Ejemplo de Co-simulación simple utilizando HELICS</b>	<b>63</b>

# Índice de Tablas

---

<b>Tabla 3-1.</b> Resumen de los principales entornos de co-simulación [6, 8].	39
<b>Tabla 3-2.</b> Simuladores más utilizados en co-simulaciones de sistemas eléctricos [6, 7, 8].	40



# Índice de Figuras

---

<b>Figura 2-1.</b> Comparativa entre "Co-Simulación" y "Simulación Combinada". Adaptada de [4].	6
<b>Figura 2-2.</b> Ejemplo de interacción de señales en una co-simulación hipotética. Adaptada de [5].	6
<b>Figura 2-3.</b> Resumen de los diferentes enfoques implicados en la co-simulación. Adaptada de [4].	7
<b>Figura 2-4.</b> Esquema relacional de componentes generales de una co-simulación. Adaptada de [11].	8
<b>Figura 2-5.</b> Tipos de modelos de simulación. Elaboración propia.	10
<b>Figura 2-6.</b> Método de sincronización maestro-esclavo: un simulador actúa como maestro [13].	13
<b>Figura 2-7.</b> Método de sincronización por pasos de tiempo [13].	14
<b>Figura 2-8 .</b> Método de sincronización global dirigido por eventos [13].	14
<b>Figura 2-9.</b> Esquema secuencial de las etapas de una co-simulación con HELICS [9].	20
<b>Figura 2-10.</b> Publicaciones por área de investigación de ‘co-simulación. Elaboración propia.	25
<b>Figura 2-11.</b> Países con mayor número de publicaciones sobre ‘co-simulación’. Elaboración propia.	26
<b>Figura 3-1.</b> Plataforma de co-simulación CyDER con aplicaciones específicas [51].	32
<b>Figura 3-2.</b> Interfaz grafica de la plataforma de co-simulación DACCOSIM-NG [52].	33
<b>Figura 3-3.</b> Logo y ejemplo de la interfaz de la plataforma de co-simulación FNCS [53].	34
<b>Figura 3-4.</b> Algunos de los simuladores más utilizados. Elaboración propia.	41
<b>Figura 4-1.</b> Esquema del entorno de co-simulación mediante HELICS para SMTD [78].	44
<b>Figura 4-2.</b> Generación de energía activa de los generadores en cada red por escenario [79].	45
<b>Figura 4-3.</b> Flujo de gas en los nodos de gas para cada red y escenario [79].	46
<b>Figura 4-4.</b> Esquema del entorno de co-simulación utilizado en la investigación [80].	47
<b>Figura 4-5.</b> Porcentaje de variación cuando se ejecuta en varios casos [80].	48
<b>Figura 4-6.</b> Impacto de DER en mercados mayorista-minorista contra la demanda [9].	49
<b>Figura 4-7.</b> Esquema del entorno de co-simulación mediante HELICS [81].	50
<b>Figura 4-8.</b> Redes de T&D utilizadas en la investigación [81, 82].	50
<b>Figura 4-9.</b> Comparación de diversos perfiles en los escenarios ejecutados [81].	51



# 1 GENERALIDADES

---

*“Todo lo puedo en Cristo que me fortalece”.*

*- Filipenses 4:13 -*

En el presente capítulo se exploran las generalidades relacionadas con la co-simulación, con un enfoque específico a los sistemas de energía eléctrica. A continuación, se ofrece una introducción que contextualiza la investigación, se describen los objetivos, la metodología empleada y se proporciona un resumen del contenido a desarrollar en este trabajo.

## 1.1 Introducción

En el contexto de la constante evolución de las tecnologías vinculadas a la generación, transmisión, distribución y almacenamiento de energía, la creciente adopción de vehículos eléctricos y la progresiva automatización de sistemas, se ha producido una transformación significativa en el paradigma de las redes de distribución eléctrica. A pesar de los innegables beneficios de estos avances, su implementación ha planteado desafíos técnicos considerables, especialmente en la integración, control y operación de sistemas energéticos cada vez más interconectados y complejos.

La simulación, entendida como una técnica de análisis de sistemas, no es simplemente una tecnología, sino un enfoque que busca complementar o expandir las experiencias reales. Su objetivo es proporcionar experiencias guiadas que repliquen aspectos sustanciales del mundo real de manera interactiva. En este contexto, la simulación se presenta como una herramienta valiosa para comprender y evaluar sistemas, permitiendo un análisis detallado y una representación dinámica de situaciones que facilitan la toma de decisiones informada.

Aunque la simulación es una de las técnicas más utilizadas en una amplia gama de campos como la ingeniería y la ciencia, ha enfrentado diversos obstáculos para su aceptación y utilidad plena. Los modelos de sistemas a gran escala suelen ser complejos, y la programación informática asociada puede resultar ardua. Además, la simulación de sistemas complejos a veces requiere un considerable tiempo de procesamiento computacional. No obstante, es crucial reconocer que las simulaciones no sustituyen el pensamiento, sino que son herramientas para comprender mejor fenómenos o sistemas complejos [1, 2].

La simulación en sistemas de energía eléctrica se ha consolidado como una metodología esencial que facilita la creación de modelos y el análisis del desempeño de las infraestructuras energéticas en un entorno virtual. Esta técnica implica la elaboración de modelos matemáticos y computacionales precisos, que representen los componentes y la dinámica de un sistema eléctrico, tales como generadores, líneas de transmisión, transformadores, cargas y sistemas de control. La simulación de redes de distribución adquiere relevancia por diversas razones: permite a los ingenieros estudiar el comportamiento de la red en condiciones difíciles de replicar en el mundo real, como eventos climáticos extremos o aumentos de demanda instantáneos; facilita la identificación y corrección de problemas, como líneas de transmisión sobrecargadas o equipos de protección

insuficientes frente a fallas; y contribuye al diseño de redes más eficientes y seguras al evaluar el impacto de diversas configuraciones en el rendimiento general. A pesar de la importancia histórica de la simulación tradicional de sistemas de energía eléctrica en la evaluación y diseño de infraestructuras eléctricas, se ha constatado su limitación para abordar la creciente complejidad. En este contexto, surge la necesidad imperante de explorar enfoques más avanzados y flexibles.

La co-simulación se presenta como una respuesta innovadora a estos desafíos, permitiendo la integración de múltiples simuladores especializados. Cada uno de estos simuladores tiene la capacidad de modelar aspectos específicos de la red eléctrica, proporcionando una representación más completa y detallada. Este enfoque no solo pretende superar las limitaciones inherentes a las simulaciones convencionales, sino que también promete proporcionar una visión más precisa de la dinámica de los sistemas de energía eléctrica contemporáneos.

La adopción de la co-simulación como herramienta presenta ventajas significativas en los sistemas de distribución eléctrica. Esto se traduce en una visión más precisa y completa al combinar simuladores especializados, generando resultados más confiables. Asimismo, permite optimizar la eficiencia y tomar decisiones informadas al observar el sistema en su conjunto. La co-simulación facilita la exploración de escenarios complejos para investigaciones avanzadas, anticipa la detección y resolución de problemas al simular situaciones reales, mejorando la confiabilidad del sistema. Además, promueve la colaboración entre expertos al permitirles contribuir con sus conocimientos en una única plataforma, fomentando la interdisciplinariedad y el intercambio de ideas.

## 1.2 Objetivos

### 1.2.1 Objetivo General

Realizar una revisión exhaustiva del estado del arte de la co-simulación en el ámbito de los sistemas de energía eléctrica, con el propósito de proporcionar una visión integral y actualizada de los aspectos teóricos y prácticos de esta técnica de análisis, modelado y simulación mediante HELICS.

### 1.2.2 Objetivos Específicos

- Exponer de manera detallada los conceptos y principios teóricos fundamentales de la co-simulación, centrándose en su aplicación específica en el contexto de sistemas de energía eléctrica.
- Determinar el estado actual de las herramientas y plataformas de co-simulación disponibles, con especial énfasis en HELICS, así como explorar su eficacia y aplicabilidad en entornos de ingeniería eléctrica.
- Realizar un análisis comparativo detallado de diversas herramientas y plataformas utilizadas para la co-simulación en sistemas de energía eléctrica, identificando sus fortalezas, limitaciones y características distintivas.
- Presentar casos de estudio relevantes que ejemplifiquen la implementación exitosa de la co-simulación en sistemas de energía eléctrica mediante HELICS, destacando los resultados obtenidos, los desafíos superados y las lecciones aprendidas.

## 1.3 Metodología

La metodología empleada para la realización de este trabajo se estructuró sistemáticamente en varios pasos claramente definidos, con el objetivo de abordar de manera integral la temática de co-simulación en sistemas de energía eléctrica. Inicialmente se llevó a cabo una exhaustiva búsqueda de información relevante sobre las herramientas y plataformas existentes para co-simulación en el ámbito de la ingeniería eléctrica. Esta fase

permitió obtener una comprensión sólida y actualizada del estado del arte en este campo.

Tras la recopilación de información, se procedió a la cuidadosa selección de las fuentes más destacadas y pertinentes, basándose en criterios como relevancia, actualidad y credibilidad. La información se organizó de acuerdo con la estructura propuesta para el trabajo, asegurando una disposición ordenada y coherente de los elementos esenciales de la investigación.

Se realizó un análisis crítico de la información obtenida, identificando las características más relevantes de las herramientas y plataformas de co-simulación. Este proceso aseguró una evaluación fundamentada de las capacidades y limitaciones de cada herramienta.

Se procedió a la síntesis de la información, destacando similitudes, diferencias y patrones emergentes entre las diversas herramientas y plataformas. Con la información sintetizada y organizada, se inició la redacción del capítulo correspondiente, asegurando una estructura clara y coherente. Esta fase proporcionó una visión holística y estructurada de los conocimientos adquiridos durante la investigación, lo que luego permitió plasmar de manera ordenada y comprensible la comparativa entre las herramientas de co-simulación.

Después de la comparativa, se incluyó un capítulo dedicado a casos de estudio y ejemplos prácticos. Este abordaje permitió ilustrar la aplicación de las herramientas y plataformas en situaciones reales, ofreciendo ejemplos concretos de co-simulaciones en el ámbito de sistemas de energía eléctrica.

Por último, después de una revisión minuciosa del trabajo, se redactaron las conclusiones y recomendaciones a considerar para futuros trabajos. Estas fueron elaboradas en base al análisis crítico y la síntesis de la información revisada, proporcionando un cierre coherente y fundamentado para el trabajo desarrollado.

## 1.4 Contenido

La estructura de este trabajo de fin de máster está organizada por diversos capítulos que se describen a continuación:

- En primer lugar, el **capítulo 1** presenta el contexto general de la co-simulación, presenta los objetivos y la metodología del trabajo, también ofrece una visión resumida del contenido de la investigación.
- En el **capítulo 2** se expone un análisis exhaustivo del estado actual de la co-simulación, abordando sus fundamentos teóricos y destacando aspectos relevantes. También se discute cómo la co-simulación puede mejorar la modelización y el análisis de los sistemas complejos.
- El **capítulo 3** se centra en comparar y analizar las principales herramientas y plataformas de co-simulación existentes, destacando especialmente HELICS. Asimismo, se presentan los simuladores más utilizados en co-simulaciones.
- En el **capítulo 4** se exploran casos de estudio y aplicaciones prácticas donde HELICS ha sido implementado de forma exitosa como herramienta o plataforma de co-simulación.
- Por último, el **capítulo 5** resume las conclusiones derivadas de la investigación, ofreciendo perspectivas clave y sugerencias para posibles desarrollos futuros en el ámbito de herramientas de co-simulación.

Se destaca que gran parte del trabajo se centró en labores de investigación, especialmente en los capítulos 2 y 3, mientras que el capítulo 4 sirve como prueba de aplicación inicial, ofreciendo una perspectiva práctica de una de las herramientas o plataformas de co-simulación seleccionadas para resaltar la viabilidad de la investigación futura.



# 2 CO-SIMULACIÓN: ESTADO DEL ARTE

---

*“Las batallas de la vida no siempre favorecen al hombre más fuerte o rápido, Pero tarde o temprano, el hombre que gana ¡Es el hombre que piensa que puede hacerlo!”.*

*- Napoleón Hill -*

En este capítulo se aborda un análisis exhaustivo y detallado del estado del arte de la co-simulación en sistemas de energía eléctrica. Se examina el panorama actual de la co-simulación desde diversas perspectivas, incluyendo aspectos teóricos, prácticos y aplicativos, con el objetivo de proporcionar una visión holística y fundamentada que sirva como base sólida para el análisis comparativo posterior. Este capítulo se establece como un punto de partida esencial para comprender la evolución y el estado actual de la co-simulación en el contexto específico de sistemas de energía eléctrica, sentando las bases para el desarrollo de la investigación y la identificación de áreas de oportunidad y mejora en este campo en constante evolución.

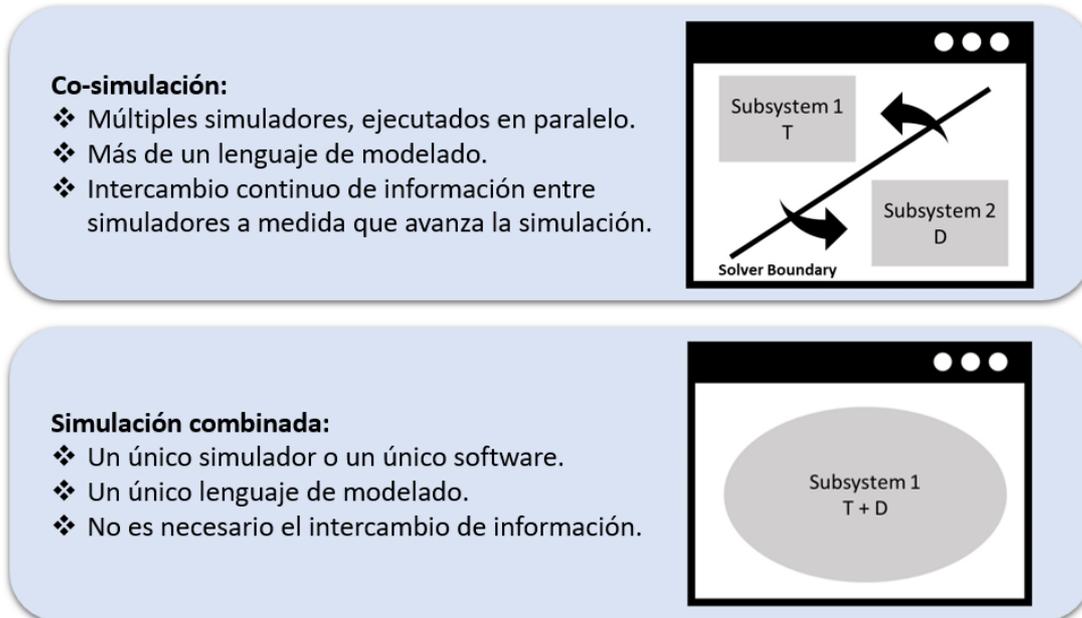
## 2.1 Concepto de ‘Co-simulación’

En la actualidad, tanto en la industria como en la ciencia, se utilizan ampliamente técnicas de modelización y simulación, pero partes de sistemas más amplios suelen modelarse y simularse con técnicas, herramientas y algoritmos distintos. Además, expertos de distintas disciplinas utilizan diversas técnicas de modelización y simulación. Ambos hechos dificultan el estudio de sistemas heterogéneos acoplados. La co-simulación es una técnica emergente que permite la simulación global de un sistema acoplado mediante la composición de las simulaciones de sus partes. Debido a su potencial y a su carácter interdisciplinario, la co-simulación se estudia en distintas disciplinas, pero los resultados se comparten de forma limitada [3].

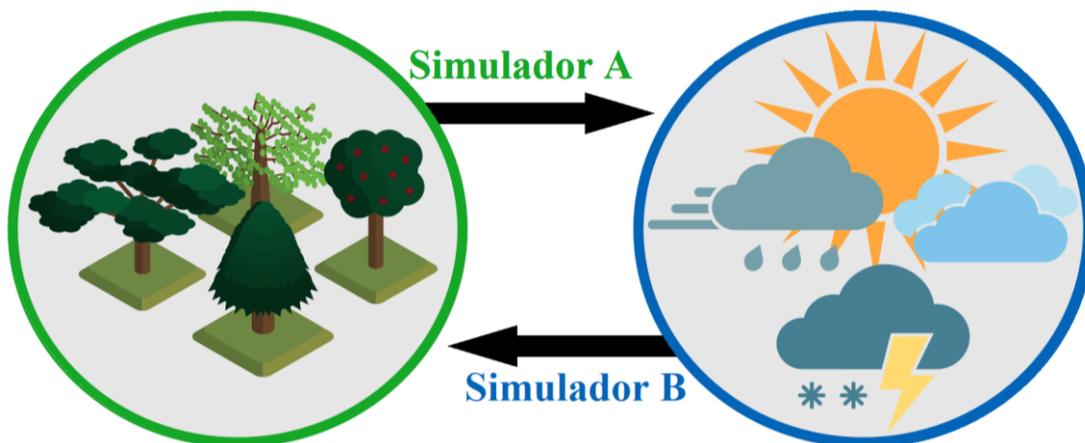
El término “co-simulación” se emplea con cierta ambigüedad y puede interpretarse de diferentes formas. En un artículo informativo de la North American Electric Reliability Corporation (NERC), se establece una diferencia entre “co-simulación” y “simulación combinada”, ver **Figura 2-1**. En la “co-simulación”, se simulan múltiples subsistemas en paralelo, donde cada uno utiliza métodos y técnicas de modelado y simulación distintos. En cambio, la “simulación combinada” implica que tanto el sistema de transmisión como el sistema de distribución se modelan y se ejecutan en la misma plataforma o interfaz. Por lo tanto, a diferencia de la co-simulación, no es necesario intercambiar información ni sincronizar las plataformas, ya que el sistema combinado se ejecuta como una entidad única [4].

Un ejemplo de “co-simulación” se puede apreciar en la **Figura 2-2**, en esta se visualiza como interactúan un simulador ‘A’ que tiene un modelo el cual describe el crecimiento de la vegetación en grandes áreas geográficas, y un simulador ‘B’ que tiene un modelo que describa las condiciones atmosféricas y el clima a medida que avanzan en el tiempo. Estos dos sistemas interactúan claramente con la precipitación y las temperaturas, que afectan al crecimiento de la vegetación, y la cantidad de vegetación, que afecta a la temperatura del aire y al contenido de humedad. En este caso, mientras que el simulador atmosférico (simulador ‘B’) puede calcular el clima actual utilizando los últimos valores de reflectividad superficial y contribuciones de humedad de la

vegetación, el simulador de vegetación (simulador 'A') puede utilizar los últimos valores del simulador atmosférico para determinar el contenido de humedad del suelo y la temperatura ambiente. En lugar de intentar construir un solo simulador que modele todas estas funcionalidades, la co-simulación nos permite utilizar simuladores existentes que tienen una inversión y validación a largo plazo detrás de ellos; esa experiencia se aprovecha para construir simulaciones de sistemas complejos de sistemas [5].



**Figura 2-1.** Comparativa entre "Co-Simulación" y "Simulación Combinada". Adaptada de [4].



**Figura 2-2.** Ejemplo de interacción de señales en una co-simulación hipotética. Adaptada de [5].

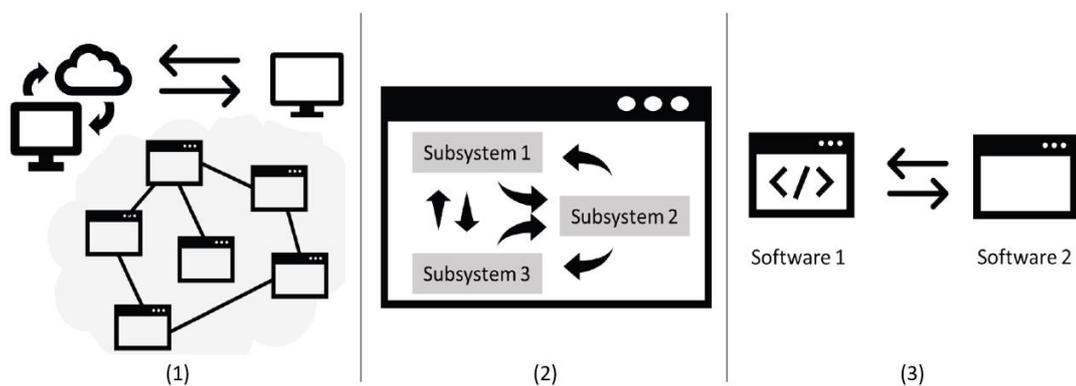
Como se ha explicado anteriormente, la co-simulación puede implicar la modelización de varios sistemas en una sola plataforma o software, o puede implicar la interfaz entre varios software o plataformas, utilizando cada uno para modelar un sistema específico. Estos enfoques se pueden clasificar en tres grupos principales:

1. **Plataforma Virtual:** Este enfoque actúa como un eje central que conecta múltiples simuladores independientes en un único ordenador o incluso en la nube. Facilita la comunicación y el intercambio de datos entre estas diversas herramientas. Ejemplo: HELICS.
2. **Plataforma Única Integrada:** En este caso, el software funciona como una plataforma única y unificada que soporta múltiples simuladores y sus respectivos entornos de modelado. Cuando se trata de herramientas altamente flexibles, esto ofrece una experiencia racionalizada, pero puede limitar la

flexibilidad. Ejemplo: DIgSILENT PowerFactory.

3. **Enlace Personalizado:** Este enfoque específico implica la creación de una interfaz personalizada que permita que dos aplicaciones de software específicas se comuniquen entre sí. Esto es ideal para simulaciones que requieren un intercambio de datos detallado entre herramientas específicas. Ejemplo: ‘TDCOSIM’ con OpenDSS y PSS/E.

En todos estos casos, crear una interfaz entre cada uno de los sistemas múltiples puede ser difícil debido a las diferencias en los detalles de modelado y la diversidad en los datos de entrada/salida que deben intercambiarse a través de las interfaces entre las diferentes plataformas [4]. Además, es difícil para estas plataformas de co-simulación lograr la escalabilidad con una velocidad de computación óptima.



**Figura 2-3.** Resumen de los diferentes enfoques implicados en la co-simulación. Adaptada de [4].

Existe un gran número de software muy desarrollados para cada uno de estos elementos en los que operan los estudios tradicionales. La co-simulación es un medio para aprovechar el software existente y encontrar una manera de capturar una imagen completa del comportamiento del sistema al proporcionar formas de comunicación entre ellos e intercambiar información valiosa.

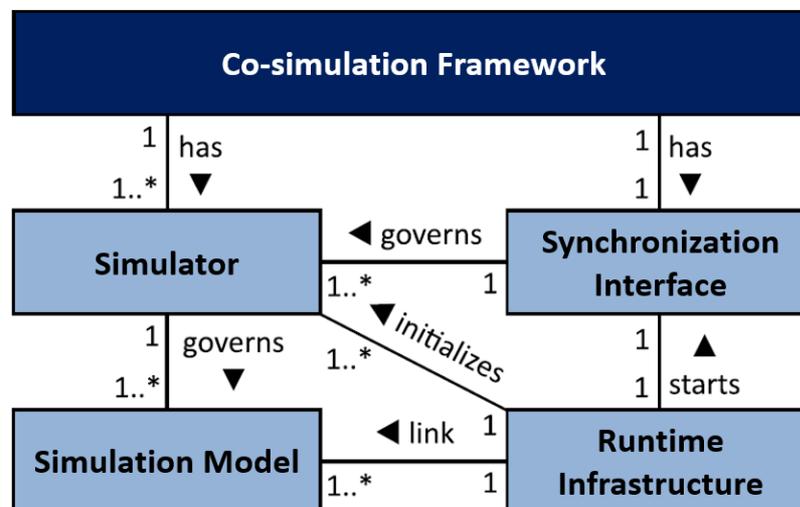
En el pasado, aunque se reconocía que los modelos agregados son aproximados (por ejemplo, modelado de carga), las capacidades tecnológicas eran insuficientes para considerar la captura de todos los detalles de modelado bajo un mismo paraguas (ya sea como un solo software o una combinación de varios). Esto, junto con el hecho de que los sistemas de transmisión, distribución y generación están volviéndose más rápidos, flexibles y variables, hace un caso convincente para seguir adelante con estudios de co-simulación para evaluar la fiabilidad y la resiliencia del sistema.

La co-simulación permite que diferentes dominios interactúen a lo largo de una simulación, intercambiando valores dinámicamente que definen condiciones límite para otros simuladores. Esto puede hacer uso de simuladores en una variedad de plataformas de cómputo y lenguajes. El interés en esta área de investigación (tanto académica como industrial) ha dado lugar al desarrollo de diversas plataformas de co-simulación. En el ámbito académico, destacan **Daccosim-NG** [6] y **EPOCHS** [7]. Daccosim-NG facilita la integración de diferentes simuladores a través de interfaces estandarizadas, mientras que EPOCHS se utiliza para la co-simulación de sistemas de energía y comunicaciones, evaluando la resiliencia y seguridad de infraestructuras críticas. Por otro lado, en el ámbito industrial, plataformas como **FNCS** [8] y **HELICS** [9] han sido desarrolladas con el apoyo de la industria y agencias gubernamentales. FNCS se enfoca en la simulación de la interoperabilidad de redes y sistemas energéticos, y HELICS es ampliamente adoptada para investigar la integración de recursos energéticos distribuidos y redes inteligentes. Estas plataformas reflejan la colaboración y la convergencia de intereses entre la academia y la industria para avanzar en la comprensión y gestión de sistemas complejos.

Básicamente, las co-simulaciones implican la integración de múltiples simuladores que operan de manera independiente y se enfocan en diferentes subsistemas o aspectos de un gran sistema complejo. Esto permite obtener resultados que facilitan la comprensión de los efectos del acoplamiento entre los subsistemas. Sin embargo, las co-simulaciones presentan desafíos relacionados con la infraestructura de ejecución y la sincronización de eventos. Los sistemas simulados pueden clasificarse en dos tipos principales: discretos o continuos (aunque también existen los sistemas híbridos), dependiendo de si las variables de estado cambian en intervalos fijos o de manera continua en el tiempo. Para modelar estos sistemas, se emplean modelos basados en eventos discretos o en tiempo continuo, según las necesidades específicas de la simulación. Por ejemplo, la consideración de la naturaleza continua de un fenómeno puede ser crucial dependiendo de los objetivos de la investigación. Una de las principales dificultades en la integración de una co-simulación radica en combinar modelos de tiempo continuo de sistemas de potencia con simulaciones de eventos discretos de redes de comunicación [10].

## 2.2 Partes de una Co-simulación

La co-simulación se basa en varios componentes clave que trabajan juntos para simular eficazmente sistemas complejos. Aunque básicamente las partes de una co-simulación pueden variar según el contexto y la aplicación específica, comúnmente las partes fundamentales de una co-simulación incluyen: los *Simuladores*, los *Modelos de simulación* a utilizar, una *Interfaz de sincronización* (orquestador) y, por último, una *Infraestructura de tiempo de ejecución* (escenario). Estos elementos son representados en la **Figura 2-4** y se detallan a continuación.



**Figura 2-4.** Esquema relacional de componentes generales de una co-simulación. Adaptada de [11].

## 2.3 Simuladores

Los **simuladores** son herramientas computacionales fundamentales en la modelización y análisis de sistemas complejos. Estos se pueden definir como algoritmos matemáticos o métodos computacionales que resuelven modelos matemáticos para imitar el comportamiento de un sistema real a través de simulaciones. Los simuladores realizan aproximaciones numéricas de soluciones a ecuaciones matemáticas complejas y realizar con precisión los cálculos necesarios para proporcionar información sobre cómo cambia un sistema con el tiempo [12].

Su función es crucial para facilitar el desarrollo de modelos dentro de su dominio especializado. Por ejemplo,

en la simulación de sistemas eléctricos, los simuladores ofrecen herramientas para describir componentes estándar utilizando parámetros físicos y definir cómo están interconectados estos componentes. A continuación, el simulador transforma estas descripciones en ecuaciones aptas para ser procesadas por el solucionador del simulador. Este enfoque permite a los simuladores aplicar métodos de modelización adaptados a las necesidades específicas de su ámbito [13]. De este modo, los simuladores pueden modelar con eficacia sistemas complejos y ofrecer una visión precisa de su comportamiento.

Los simuladores son versátiles y flexibles, capaces de manejar diferentes tipos de modelos y escenarios, lo que les permite abordar una amplia gama de situaciones de simulación. También están diseñados para ser eficientes y escalables, gestionando la complejidad computacional de los modelos, especialmente en sistemas a gran escala o con numerosos parámetros y variables. Algunos simuladores ofrecen capacidades de visualización para representar gráficamente los resultados de la simulación y generar informes para comunicar los resultados a las partes interesadas.

Las aplicaciones de los simuladores abarcan un amplio espectro de disciplinas. En ingeniería, se utilizan para simular circuitos electrónicos, estructuras mecánicas y sistemas de flujo de fluidos. En ciencias, los simuladores modelan la dinámica del cambio climático, la propagación de enfermedades infecciosas y la evolución de los ecosistemas. En el ámbito empresarial, se utilizan simuladores para analizar cadenas de suministro, optimizar procesos de producción y evaluar estrategias de marketing. En las ciencias sociales, los simuladores se emplean para estudiar el comportamiento de la población humana, modelar las interacciones sociales y evaluar el impacto de las intervenciones políticas.

El requisito fundamental de cualquier simulador es la reproducción del comportamiento del sistema dentro de un entorno físico lo más realista posible desde la perspectiva de un operador. Aunque el simulador incorpore algunas características físicas del sistema, los componentes sustanciales del mismo sólo existen necesariamente en forma de modelos [14]. Gracias a los simuladores, los investigadores e ingenieros pueden analizar sistemas complejos, explorar distintos escenarios, optimizar el rendimiento del sistema y tomar decisiones fundamentadas basándose en los resultados de la simulación.

## 2.4 Modelos de Simulación

Un **modelo de simulación** es una representación abstracta de un sistema, proceso o fenómeno del mundo real que se crea con el propósito de estudiar, analizar o predecir su comportamiento. Este modelo se desarrolla utilizando técnicas matemáticas y herramientas de simulación para imitar las operaciones y características del sistema real [2]. Los modelos de simulación son como versiones matemáticas de situaciones del mundo real. Pueden representar cualquier cosa, desde cómo funciona una red eléctrica hasta cómo se comportan las redes de comunicación.

Los modelos de simulación pueden variar en complejidad y enfoque dependiendo del sistema que se esté analizando y los objetivos de la simulación. Según se explica en [2], los modelos de simulación se pueden clasificar fácilmente en tres dimensiones diferentes, estas son: Temporalidad, Incertidumbre y Naturaleza del cambio. En la **Figura 2-5** se puede apreciar un diagrama con los diferentes tipos de modelos de simulación clasificados según la dimensión a la cual pertenecen, estas se describen a continuación.

### 2.4.1 Modelos de Simulación según su Temporalidad

Esta clasificación se refiere a cómo se modela y representa el tiempo en la simulación. Incluye la distinción entre modelos estáticos y dinámicos, que se centran en instantáneas específicas del sistema versus la evolución del sistema a lo largo del tiempo, respectivamente.

- Los *Modelos Estáticos* representan un sistema en un momento específico o en un período donde el tiempo no juega un papel importante. Estos modelos se enfocan en una "foto instantánea" del sistema en un momento particular y no capturan su evolución a lo largo del tiempo. Estos modelos tienden a ser

menos complejos computacionalmente, ya que no requieren cálculos continuos o iterativos a lo largo del tiempo. Los modelos estáticos son apropiados cuando la atención se centra en comprender el estado actual del sistema o en evaluar los posibles resultados en diferentes escenarios sin tener en cuenta la influencia del tiempo. Ejemplo: Modelos de análisis de redes eléctricas y modelos de inventario de una empresa.

- Los *Modelos Dinámicos* capturan el comportamiento de un sistema a medida que evoluciona con el tiempo. Consideran los cambios en las variables del sistema y las interacciones a lo largo del tiempo, permitiendo estudiar la dinámica del sistema y el impacto de factores dependientes del tiempo en su comportamiento. Estos modelos pueden ser más complejos computacionalmente, especialmente si implican la resolución de ecuaciones diferenciales o la simulación de eventos a lo largo del tiempo. Los modelos dinámicos son esenciales cuando el sistema muestra un comportamiento dinámico y el objetivo es seguir su evolución en el tiempo, analizar los bucles de retroalimentación u optimizar el rendimiento en condiciones cambiantes. Ejemplo: Modelos de tráfico urbano y modelos de crecimiento poblacional.

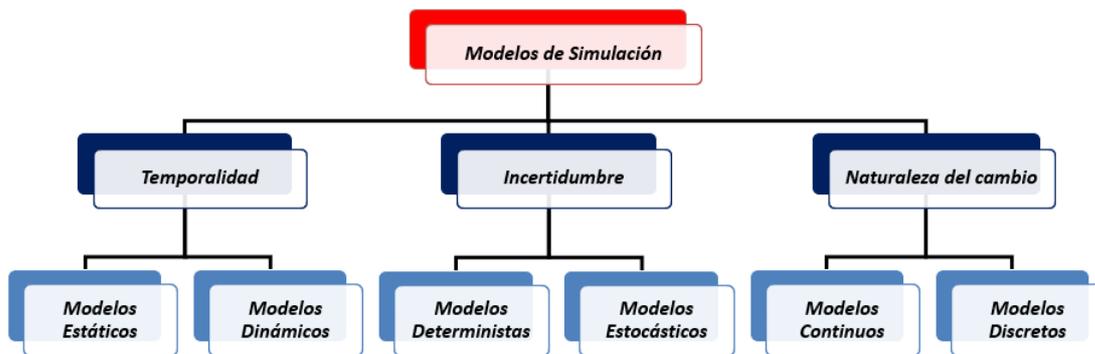


Figura 2-5. Tipos de modelos de simulación. Elaboración propia.

## 2.4.2 Modelos de Simulación según su Incertidumbre

Esta clasificación aborda la presencia o ausencia de aleatoriedad e incertidumbre en el modelo de simulación. Se divide en modelos deterministas y estocásticos, dependiendo de si el modelo incluye o no elementos aleatorios y probabilísticos.

- Los *Modelos Deterministas* son aquellos que no incluyen elementos aleatorios o incertidumbre en el modelo. La salida de un modelo determinista está completamente determinada por los parámetros de entrada y la estructura del modelo. Estos modelos proporcionan un resultado único y predecible para un conjunto dado de entradas. Los modelos deterministas son adecuados para sistemas con relaciones bien definidas entre variables e influencias aleatorias mínimas, en los que el objetivo es predecir resultados específicos con un alto grado de precisión. Ejemplo: Modelado de la trayectoria de proyectiles y modelados para predicción de tendencias económicas a partir de datos históricos.
- Los *Modelos Estocásticos* incorporan elementos aleatorios y probabilísticos en el modelo. Estos modelos tienen en cuenta la incertidumbre en el sistema introduciendo variables aleatorias que influyen en su comportamiento. Debido a la naturaleza aleatoria de los parámetros de entrada, los modelos estocásticos producen resultados diferentes para cada ejecución de la simulación. Los modelos estocásticos son indispensables cuando la incertidumbre desempeña un papel importante en el comportamiento del sistema y el objetivo es comprender la distribución de los posibles resultados, evaluar los riesgos o valorar el impacto de los factores aleatorios. Ejemplo: Modelos de simulaciones de Montecarlo y modelos de análisis de propagación de enfermedades.

### 2.4.3 Modelos de Simulación según Naturaleza del Cambio

Esta Clasificación se refiere a cómo se representan los cambios en el sistema durante la simulación. Incluye la distinción entre modelos continuos y discretos, que se centran en cambios suaves y continuos versus cambios discretos y discontinuos en el sistema.

- Los *Modelos Continuos* son aquellos que representan sistemas donde las variables pueden cambiar de manera infinitesimal en cualquier momento dentro de un intervalo de tiempo definido. Estos modelos utilizan ecuaciones diferenciales, integrales u otras funciones matemáticas para describir el comportamiento del sistema, las relaciones entre variables y sus tasas de cambio. Estas ecuaciones se resuelven numéricamente para obtener la evolución del sistema. La simulación continua es adecuada para estudiar procesos que involucran cambios suaves y continuos a lo largo del tiempo. Ejemplo: Modelización del cambio climático, la dispersión de la contaminación atmosférica o la gestión de los recursos hídricos.
- Los *Modelos Discretos* representan sistemas en los que las variables de estado cambian instantáneamente en distintos momentos debido a sucesos discretos. Los cambios en el estado del sistema se producen en momentos específicos, como llegadas o salidas, que desencadenan transiciones de estado. El sistema evoluciona a través de una secuencia de eventos, con variables de estado que cambian sólo en estos momentos de eventos, que son contables. Este método de modelización es adecuado para sistemas basados en sucesos, centrándose en su secuencia e impacto. Ejemplo: En redes de comunicaciones para el modelado del comportamiento de los paquetes de datos en una red, donde los paquetes se transmiten y reciben en puntos discretos en el tiempo.

Comprender estas diferencias es crucial para seleccionar el enfoque de modelado de simulación apropiado. Esta elección depende de las características del sistema que se estudia, el nivel de detalle requerido y los objetivos específicos de investigación del estudio de simulación. En el contexto de la simulación, la mayor parte de los modelos suelen ser estocásticos y dinámicos. Por ejemplo, los sistemas eléctricos suelen simularse con ecuaciones algebraicas y diferenciales, mientras que las redes de comunicación se modelan como sistemas de eventos discretos, donde los cambios ocurren en momentos específicos y no duran ningún tiempo real [12].

## 2.5 Infraestructura de Tiempo de Ejecución

La **infraestructura en tiempo de ejecución** (también llamada '*infraestructura de comunicación*') hace referencia a los mecanismos y arquitecturas que se encargan de coordinar e intercambiar la información en la simulación. En algunos casos, todas las herramientas pueden estar en el mismo ordenador, compartiendo recursos y comunicándose fácilmente (como un equipo que trabaja en la misma oficina), esto se denomina una arquitectura de *simulación paralela*. Pero en otros casos, las herramientas pueden estar repartidas en distintos ordenadores situados en diferentes lugares, lo que puede hacer que la comunicación sea más lenta y complicada (como un equipo que trabaja a distancia desde distintas ciudades), esto se conoce como arquitectura de *simulación distribuida*. Estas infraestructuras pueden no ser necesarias en todas las configuraciones de simulación, pero en el ámbito de las co-simulaciones son muy importantes, esto debido a que es el componente responsable de garantizar que todos los simuladores estén sincronizados en el tiempo y que estén intercambiando datos en el momento adecuado [10, 12].

### 2.5.1 Infraestructura de Simulación Paralela

La **infraestructura de simulación paralela** se puede definir como un enfoque computacional que utiliza múltiples procesadores o unidades de procesamiento para ejecutar simultáneamente diferentes partes de un modelo de simulación. En lugar de ejecutar el modelo secuencialmente en un solo procesador, la simulación paralela divide el modelo en secciones que pueden ejecutarse de manera independiente en varios procesadores

al mismo tiempo. Este enfoque permite una ejecución más rápida de la simulación al distribuir la carga de trabajo entre múltiples procesadores, lo que resulta en tiempos de simulación más cortos y una mayor eficiencia computacional [12, 15].

En la simulación paralela, cada procesador o nodo de procesamiento puede ejecutar una parte independiente de la simulación, procesando datos y realizando cálculos de manera simultánea. Luego, los resultados parciales se combinan al final para obtener el resultado final de la simulación. Este enfoque es especialmente útil para modelos de simulación complejos que requieren una gran cantidad de cálculos y pueden beneficiarse de la capacidad de procesamiento paralelo ofrecida por sistemas informáticos multiprocesador o distribuidos.

Este enfoque puede implementarse de diferentes maneras, incluidas las arquitecturas de memoria compartida y distribuida. En la *memoria compartida*, todos los procesadores tienen acceso a una única memoria central, lo que facilita la comunicación entre los procesadores, pero puede generar cuellos de botella en el rendimiento si muchos procesadores intentan acceder a la memoria al mismo tiempo. En la *memoria distribuida*, cada procesador tiene su propia memoria y se comunica con otros procesadores a través de mensajes, lo que puede ser más escalable, pero puede requerir un mayor esfuerzo de programación para coordinar la comunicación entre procesadores.

### 2.5.2 Infraestructura de Simulación Distribuida

La **infraestructura de simulación distribuida** se refiere a un enfoque en el que diferentes partes de un modelo de simulación se ejecutan en sistemas informáticos separados, generalmente conectados a través de una red de comunicación. A diferencia de la simulación paralela, donde múltiples procesadores trabajan juntos en una sola máquina para ejecutar diferentes secciones del modelo, en la simulación distribuida, cada parte del modelo se ejecuta en una máquina diferente. Estas máquinas pueden estar ubicadas en diferentes lugares geográficos y pueden ser sistemas independientes con su propio hardware y software [12, 15].

En la simulación distribuida, la comunicación entre las diferentes partes del modelo se realiza a través de la red, utilizando protocolos de comunicación específicos. Cada parte del modelo puede enviar y recibir datos relevantes para la simulación a otras partes del sistema distribuido, lo que permite una colaboración y coordinación eficientes entre los diferentes componentes del modelo.

Esta arquitectura es especialmente útil cuando se requiere la integración de sistemas heterogéneos o la colaboración entre múltiples equipos de investigación que trabajan en diferentes ubicaciones. Permite la ejecución de modelos de simulación complejos que pueden abarcar una amplia gama de escenarios y condiciones, aprovechando los recursos informáticos disponibles en diferentes lugares. Además, la simulación distribuida puede ofrecer escalabilidad y redundancia, lo que la hace adecuada para aplicaciones que requieren un alto rendimiento y confiabilidad.

La infraestructura de tiempo de ejecución facilita la gestión de datos durante el tiempo de ejecución de la simulación, incluido el procesamiento de datos de entrada, la recopilación de datos de salida, el almacenamiento y la recuperación [2]. Proporciona el entorno de ejecución donde se ejecuta el modelo de simulación gestionando la asignación de recursos informáticos, memoria y potencia de procesamiento necesarios para simular el sistema. Además, permite el control y supervisión del proceso de simulación, incluyendo funciones como pausar, reanudar o finalizar simulaciones, y monitorizar el rendimiento y progreso. Las co-simulaciones plantean varios retos en cuanto a la infraestructura de tiempo de ejecución adoptada y el modo en que se sincronizan los eventos.

## 2.6 Interfaz de Sincronización

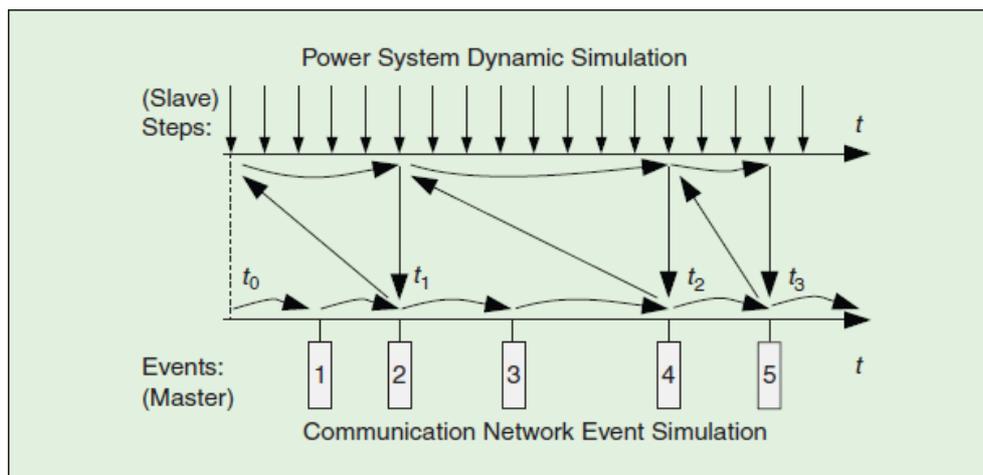
La **sincronización** de la simulación consiste en el método para coordinar la sincronización y los eventos entre diferentes simuladores para garantizar que funcionan de forma sincronizada en el entorno de simulación

conjunta. Se utilizan distintos métodos y enfoques de sincronización para gestionar las interacciones entre simuladores y garantizar que los resultados de la simulación sean coherentes y precisos. Al sincronizar los tiempos de los eventos y las acciones entre varios simuladores, los investigadores pueden crear simulaciones más realistas y completas [10]. Tradicionalmente se han utilizado dos tipos principales de mecanismos de sincronización: la sincronización conservadora y la optimista.

- En la *sincronización conservadora*, el objetivo principal es cumplir estrictamente la restricción de causalidad local. Esto significa garantizar que ningún evento se procesa antes de su hora designada, evitando violaciones de la causalidad. Sin embargo, la sincronización conservadora tiene inconvenientes: limita el paralelismo en las simulaciones al requerir el procesamiento secuencial de eventos que podrían interactuar, y puede ser sensible incluso a pequeños cambios en el modelo, lo que provoca problemas de rendimiento [12].
- Por otro lado, la *sincronización optimista* permite violaciones de la restricción de causalidad local, pero incluye mecanismos para detectar y recuperarse de estas violaciones. En la sincronización optimista, cada proceso lógico (LP) avanza en su simulación de forma independiente sin esperar mensajes de otros procesadores, lo que permite una mayor flexibilidad y un rendimiento potencialmente mejor en comparación con la sincronización conservadora [2].

En el contexto de la co-simulación, la sincronización entre varios simuladores es un desafío importante. Los tres principales mecanismos de sincronización son: maestro-esclavo, por pasos de tiempo y global dirigido por eventos. A continuación, se explican estos de manera detallada y utilizando como ejemplo una co-simulación entre un simulador de un sistema eléctrico y un simulador de redes de comunicación.

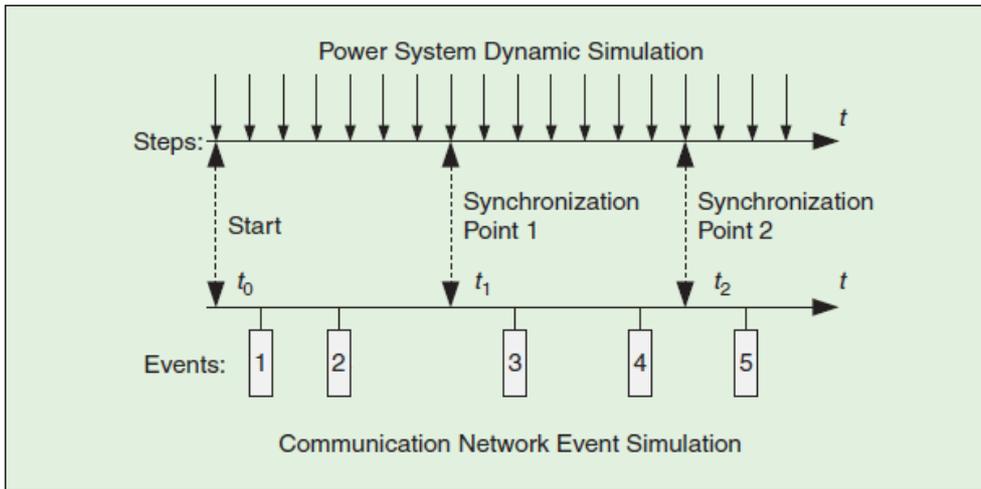
- En el *método maestro-esclavo*, un simulador es designado como el maestro y coordina todas las etapas de la co-simulación. Este maestro tendrá la máxima prioridad durante la ejecución y coordinará todos los pasos de la co-simulación. Este método se representa en la **Figura 2-6; Error! No se encuentra el origen de la referencia.**, donde durante toda la simulación, el rol del maestro es desempeñado por el simulador de comunicación mientras que el simulador del sistema eléctrico es el esclavo controlado por el maestro. Se puede observar que el maestro inicia la simulación y se ejecuta siempre y cuando no necesite datos del esclavo (desde  $t_0$  hasta  $t_1$  en la figura). Una vez que necesita los datos, el esclavo toma el control y se simula desde  $t_0$  hasta  $t_1$ , luego envía la información al maestro [13, 16].



**Figura 2-6.** Método de sincronización maestro-esclavo: un simulador actúa como maestro [13].

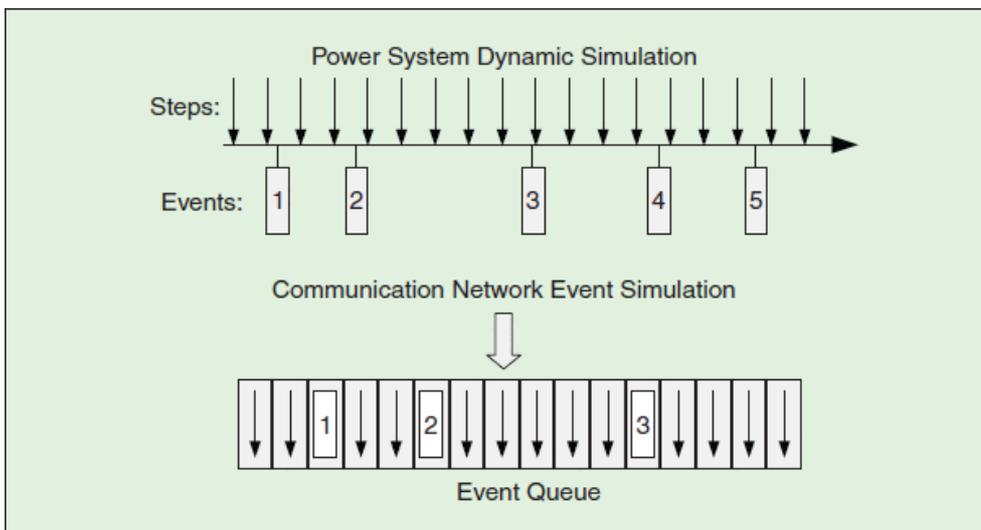
- En el *método por pasos de tiempo*, cada simulador corre independientemente y se detiene en puntos de sincronización predefinidos para comunicarse con otros simuladores. Como se muestra en la **Figura 2-7**, los simuladores individuales se ejecutan en paralelo y se detienen en los puntos de sincronización

para intercambiar información. En este caso, los simuladores se ejecutan en paralelo, por lo que los componentes de la co-simulación deben tener esta capacidad distribuida. Los puntos de sincronización están predeterminados. El problema con este método es que puede introducir imprecisiones en la co-simulación, esto debido a que cuando las variables de salida del sistema necesitan ser intercambiadas entre dos puntos de sincronización, ambos simuladores tienen que esperar hasta el próximo punto de sincronización. Este retraso introduce la acumulación de errores en la simulación y posiblemente afecta la precisión de los resultados generales de la simulación. [13, 16].



**Figura 2-7.** Método de sincronización por pasos de tiempo [13].

- En el *método global dirigido por eventos*, las iteraciones de la simulación de sistemas de potencia continuos se consideran eventos discretos y se mezclan con los eventos de la red de comunicación en una lista de eventos global, como se puede apreciar en la **Figura 2-8**. En este mecanismo, las rondas de iteración de la simulación continua del sistema de energía se consideran como eventos discretos y se mezclan con los eventos de la red de comunicación en una lista de eventos global. Esta lista forma una cola de eventos y se ordena según la marca de tiempo de estos diferentes eventos. En este tipo de mecanismo, los procesos de eventos deben ejecutarse uno tras otro y un programador de eventos global se encarga de la cola dando paso a los simuladores cuyo turno es ejecutarse [13, 16].



**Figura 2-8.** Método de sincronización global dirigido por eventos [13].

La elección del método de sincronización depende de varios factores, como la precisión deseada, el rendimiento, la capacidad de ejecución paralela de los simuladores y la estructura de la plataforma de co-simulación utilizada. Es importante seleccionar el método más adecuado para garantizar la precisión y eficiencia de la simulación, así como para evitar errores de sincronización que puedan afectar la validez de los resultados obtenidos.

## 2.7 Tipos de simulaciones

Existen diferentes tipos de simulaciones, que pueden categorizarse según la complejidad de los modelos explicados anteriormente y la arquitectura de la simulación. La elección del tipo de simulación adecuado depende de la naturaleza del sistema que se está modelando, los objetivos específicos del análisis y las características del sistema que se desean capturar. Cada tipo de simulación tiene sus propias ventajas y limitaciones, y la selección del enfoque adecuado requiere una comprensión profunda del sistema y los resultados deseados. En el campo de la co-simulación, los sistemas simulados básicamente pueden ser discretos, continuos o híbridos.

Para modelar estos sistemas, se pueden utilizar modelos basados en eventos discretos y en tiempo continuo, dependiendo de las necesidades de las simulaciones (por ejemplo, si considerar la naturaleza continua de un fenómeno puede ser importante en función de los objetivos de la investigación). Una de las principales cuestiones en la integración en una co-simulación es combinar modelos de tiempo continuo de sistemas de potencia con simulaciones de eventos discretos de redes de comunicación [10].

### 2.7.1 Simulaciones de Eventos Discretos

Están compuestas por simuladores que modelan un sistema a medida que evoluciona considerando los cambios de estado de las variables en puntos específicos del tiempo, donde un evento es un suceso que puede modificar el estado del sistema [10].

Algunas de las características esenciales de un sistema eventos discretos incluyen: reactividad (reacción instantánea a estímulos externos, como apagarlo por una entidad externa) y transitoriedad (un sistema eventos discretos puede cambiar su estado múltiples veces en el mismo punto de tiempo simulado y recibir estímulos simultáneos). Estas características se aplican en la co-simulación basada en eventos discretos, donde el orquestador reconoce que las unidades de simulación pueden evolucionar su estado interno e intercambiar valores a pesar de que el tiempo simulado está detenido.

Sin embargo, existen desafíos en la implementación de la co-simulación basada en eventos discretos. Uno de ellos es la causalidad, para evitar violaciones de esta, se requiere un orden global en los eventos intercambiados entre las unidades de simulación. Los algoritmos de orquestación optimistas pueden ser más rápidos en términos de tiempo de reloj, pero requieren capacidades de retroceso de las unidades de simulación, lo que puede ser complejo y costoso en términos de recursos computacionales [3].

Otro desafío es el determinismo y la confluencia. Es importante asegurar que la co-simulación produzca resultados deterministas y que los diferentes caminos de ejecución conduzcan al mismo resultado. Esto puede ser difícil de lograr, especialmente en sistemas de eventos discretos de caja negra. La estructura dinámica es otro desafío. Las dependencias entre las unidades de simulación pueden cambiar con el tiempo, lo que requiere flexibilidad en la estructura de la co-simulación.

Finalmente, la distribución geográfica de las unidades de simulación puede introducir desafíos adicionales, como costos de comunicación más altos y problemas de seguridad. Se han propuesto diversas soluciones para abordar estos desafíos, incluida la asignación de la computación [17] [18], el encapsulamiento jerárquico [19] y el uso de modelos de navegación muerta [20].

### 2.7.2 Simulaciones de Tiempo Continuo

Este es un enfoque de modelización que pretende integrar múltiples modelos de simulación que operan en escalas de tiempo continuas para analizar de forma exhaustiva sistemas con dinámicas continuas. Permite a los investigadores conocer mejor las interacciones y dependencias entre los distintos componentes del sistema a medida que evolucionan continuamente en el tiempo. Este método es especialmente útil para estudiar sistemas en los que las variables cambian de forma suave y continua, como los procesos físicos, los flujos de fluidos y los sistemas dinámicos [10].

Este enfoque permite interacciones dinámicas entre modelos de subsistemas, lo que posibilita la simulación de sistemas complejos en los que los componentes se influyen mutuamente en tiempo real. La co-simulación requiere la sincronización del tiempo de simulación en todos los modelos para garantizar la coherencia de las interacciones y el intercambio de datos entre subsistemas. Para mantener la sincronización se utilizan pasos temporales y métodos de integración.

Diversas herramientas y plataformas de simulación admiten la co-simulación basada en el tiempo continuo, proporcionando capacidades para integrar y simular modelos continuos de diferentes dominios de forma coordinada. Este enfoque tiene aplicaciones en ámbitos como la automoción, la industria aeroespacial, la robótica y los sistemas de energías renovables, donde la comprensión de las interacciones entre componentes continuos es fundamental para el rendimiento y la optimización del sistema.

Entre los retos que plantea la co-simulación en tiempo continuo figuran la composición modular y el tratamiento de las restricciones algebraicas. Al acoplar subsistemas pueden producirse desajustes en las entradas y salidas, lo que requiere soluciones como el Coordinador de Condiciones Límite (BCC, por sus siglas en inglés). Además, los bucles algebraicos, en los que las variables dependen indirectamente de sí mismas, plantean retos que se mitigan mediante técnicas como la iteración de punto fijo para la convergencia. La estabilidad implica garantizar que el sistema permanezca acotado en el tiempo, afectado por distintos métodos de acoplamiento y funciones de aproximación. Mantener la continuidad, especialmente con cambios repentinos en las entradas, requiere métodos de interpolación extrapolada (métodos que ayudan a estimar valores entre puntos conocidos y más allá de esos puntos) para asegurar una transición suave y continua. Cumplir las restricciones del tiempo real implica garantizar el funcionamiento eficaz de todas las unidades de simulación, con técnicas como el filtrado de Kalman que gestiona el ruido en las señales y algoritmos de orquestación que compensan los retrasos de la red [3].

La co-simulación basada en el tiempo continuo proporciona un potente método para estudiar el comportamiento dinámico de sistemas complejos compuestos por componentes continuos. Mediante la integración de múltiples modelos continuos y la simulación de sus interacciones a lo largo del tiempo, los analistas pueden obtener información sobre el rendimiento, el comportamiento y las interacciones de los sistemas, esencial para el diseño, la optimización y la toma de decisiones en diversos ámbitos de la ingeniería y la ciencia.

### 2.7.3 Simulaciones Híbridas

La co-simulación híbrida es un enfoque innovador que combina modelos de simulación de eventos continuos y discretos para analizar de forma exhaustiva sistemas con dinámicas mixtas [10]. Al integrar modelos continuos que representan comportamientos continuos con modelos de eventos discretos que capturan eventos en puntos temporales específicos, la co-simulación híbrida ofrece una representación más realista de sistemas complejos, por lo tanto, proporciona una potente herramienta para analizar el comportamiento, el rendimiento y la optimización de sistemas en escenarios reales.

Esta técnica de modelización integra modelos que operan en diferentes escalas temporales, lo que permite representar con precisión sistemas cuyos componentes presentan comportamientos tanto continuos como

discretos. Facilita la interacción entre componentes continuos y discretos, garantizando la sincronización del tiempo de simulación y permitiendo intercambios dinámicos de información.

La co-simulación híbrida tiende un puente entre los comportamientos continuos y discretos, ofreciendo una integración perfecta de elementos contrastados dentro de un único entorno de simulación. Mejora las capacidades de modelado de sistemas, la flexibilidad y la aplicabilidad a ámbitos científicos y de ingeniería complejos.

Algunos de los retos que enfrenta este tipo de simulación incluye la adaptación semántica, los tamaños de paso predictivos, la localización de eventos, la identificación de discontinuidades, los bucles algebraicos, el comportamiento de Zenón y la teoría de estados aproximados de eventos discretos. Las normas o estándares para la co-simulación híbrida, como la '*Functional Mockup Interface (FMI)*' y la '*High Level Architecture (HLA)*', siguen evolucionando con propuestas de ampliaciones y casos de prueba idealizados para garantizar enfoques sólidos [3]. Tanto la FMI como la HLA serán descritos en la siguiente sección de este capítulo.

## 2.8 Estándares de Co-simulación

Los **estándares de co-simulación** son fundamentales para la integración efectiva de diversos métodos y herramientas de co-simulación, los cuales provienen de diferentes campos de aplicación. Esta diversidad ha generado la necesidad de unificación, lo cual se persigue mediante la especificación de estándares. Sin embargo, estos estándares están en constante revisión por parte de los desarrolladores y también son extendidos por otros investigadores para satisfacer requisitos específicos. Dos de los estándares más populares, ampliamente encontrados en la literatura revisada, son la '*High Level Architecture (HLA)*' y la '*Functional Mockup Interface (FMI)*'. La HLA proporciona una interfaz y un conjunto de pautas para dirigir la ejecución de una co-simulación. Por otro lado, la FMI ofrece una interfaz para la comunicación con los modelos de simulación. Ambos estándares son ampliamente utilizados y ofrecen características de reutilización y generalidad que son altamente valoradas. Ambos estándares se describen con más detalles a continuación.

### 2.8.1 High Level Architecture (HLA)

El HLA (IEEE 1516) es una especificación de arquitectura de software diseñada para abordar la necesidad de reutilización e interoperabilidad de simulaciones dentro del Departamento de Defensa de los Estados Unidos. Proporciona un marco que define elementos funcionales, interfaces y reglas de diseño para aplicaciones de simulación, así como un marco común para la definición de arquitecturas específicas del sistema. La HLA es independiente del software y del lenguaje de programación, lo que la hace versátil y adaptable a diversos entornos de simulación [21].

Sus componentes principales son los federados, la infraestructura en tiempo de ejecución (RTI) y la interfaz en tiempo de ejecución, que permiten la interacción entre los federados y la coordinación de la federación. Los federados pueden ser diversos, desde simulaciones por computadora hasta simuladores tripulados e interfaces con jugadores en vivo, siempre y cuando permitan la interacción con otros objetos a través de intercambios de datos a través de los servicios del RTI. La infraestructura en tiempo de ejecución proporciona servicios para la coordinación del flujo de datos entre los componentes de la arquitectura, garantizando la sincronización adecuada entre ellos [16]. En una federación HLA, cada simulador participante se conoce como federado y se comunica con otros federados a través del RTI, que actúa como una instancia de administración central [10]. El RTI proporciona servicios necesarios para coordinar las operaciones y el intercambio de datos entre los federados durante la ejecución de la simulación, facilitando así la sincronización de las simulaciones y el enrutamiento de las comunicaciones entre los federados.

La HLA ofrece ventajas significativas, como escalabilidad para realizar simulaciones altamente paralelas de sistemas a gran escala, así como flexibilidad, interoperabilidad y reutilización de modelos en diferentes

contextos. Sin embargo, su implementación puede presentar desafíos, especialmente cuando se trata de hacer que los simuladores existentes sean compatibles con las especificaciones de HLA, lo que puede requerir modificaciones significativas en el código fuente existente [16].

### 2.8.2 Functional Mockup Interface (FMI)

El FMI es un estándar para el intercambio de modelos y la co-simulación, iniciado por el proyecto MODELISAR y actualmente mantenido y desarrollado por la Modelica Association. Este estándar define la forma en que se deben construir las Unidades de Maqueta Funcional (Functional Mockup Units, FMUs) para que puedan ser importadas por herramientas que actúan como orquestadores maestros y las funcionalidades e interfaces para estos últimos [21].

Cuando se exporta una FMU para intercambio de modelos, la herramienta donde se ha implementado el modelo respectivo lo traduce en un modelo de sistema dinámico en código C con entradas y salidas. Los modelos pueden contener eventos, así como ecuaciones diferenciales, algebraicas o discretas. En el caso de la FMI para co-simulación, no solo se incluye el modelo, sino también el algoritmo de solución en el código exportado [21]. Las FMUs pueden contener solucionadores, lo que permite que operen en dos modos: proporcionar acceso a las ecuaciones del modelo o implementar un solucionador que manipula las ecuaciones del modelo. Esta flexibilidad permite realizar simulaciones en paralelo y de manera distribuida, similar a la Arquitectura de Alto Nivel (HLA), lo que puede resultar en simulaciones más rápidas que en casos de un solo hilo [16].

La FMI unifica las interfaces para el intercambio de modelos y la co-simulación en su versión actual (FMI 2.0). Además, incluye características adicionales, como la posibilidad de obtener y establecer un estado de FMU (potencialmente permitiendo retroceso) y la inclusión de dependencias de entrada y salida de variables y sus derivadas (importante para la detección de bucles algebraicos). A pesar de sus ventajas, la implementación de la FMI puede presentar desafíos debido a las características opcionales del estándar, muchas de las cuales no son compatibles con la mayoría de las herramientas (especialmente de código abierto). Esto puede dificultar la implementación de métodos de acoplamiento que requieran funcionalidades específicas, como el retroceso del simulador, información sobre derivadas o dependencias de entrada y salida [16, 21].

### 2.8.3 Ad-hoc Architectural Style

En este enfoque, la arquitectura puede adaptarse a los requisitos específicos de la co-simulación, incluidos los tipos de simuladores utilizados, la complejidad del sistema modelado y el nivel deseado de comunicación y sincronización entre los simuladores. Los desarrolladores tienen la libertad de elegir las herramientas y tecnologías que mejor se adapten a las necesidades del proyecto. Esto puede implicar el uso de marcos de co-simulación existentes o el desarrollo de componentes personalizados desde cero [22].

Una característica clave de este enfoque es que la arquitectura puede no ser fácilmente reutilizable para otros proyectos de co-simulación porque está altamente personalizada. Sin embargo, las lecciones aprendidas y los patrones de diseño desarrollados durante el proyecto pueden ser valiosos para futuros esfuerzos de co-simulación. El enfoque ad-hoc también puede implicar la modificación directa de las herramientas de simulación para hacerlas compatibles entre sí. Esto puede lograrse mediante la implementación de interfaces entre los simuladores dentro de ellos mismos, en lugar de utilizar un orquestador de co-simulación [22]. Si bien este método puede ser conciso y eficiente, carece de un orquestador que gestione las interacciones y la interdependencia entre los simuladores, lo que puede llevar a resultados inexactos si no se configura adecuadamente.

Una desventaja de este enfoque es que puede requerir adaptaciones profundas a las herramientas utilizadas. Configuraciones de interfaz específicas del simulador pueden dificultar la reutilización de un simulador en otro proyecto de co-simulación sin adaptaciones significativas. Sin embargo, la adopción del estándar FMI puede ayudar a mitigar este problema, al proporcionar una interfaz estándar que puede ser utilizada tanto en enfoques

ad-hoc como en modo de algoritmo maestro [16].

Como se ha explicado anteriormente, los estándares de co-simulación, incluidos HLA y FMI estándar, así como enfoques ad-hoc, son responsables del intercambio de datos y la sincronización entre diferentes simuladores. Desde el punto de vista de la arquitectura, estos marcos pueden seguir varios estilos para estructurar componentes. Una forma común es tener un componente central que se encargue de los problemas de sincronización, como en el caso de HLA, aunque también son posibles otros enfoques de gestión, como los modelos federados típicos de la Arquitectura de Alto Nivel (HLA). Sin embargo, los enfoques ad-hoc ofrecen flexibilidad en la adaptación a necesidades específicas del proyecto, aunque pueden no ser tan reutilizables como los estándares como HLA y FMI.

## 2.9 Etapas Generales de la Co-simulación

La co-simulación es un proceso complejo que combina múltiples simuladores individuales para modelar un sistema complejo de forma integral. Este proceso implica varias etapas clave, las cuales se pueden agrupar en cuatro fases principales: *creación*, *inicialización*, *ejecución* y *estado final*. Estas etapas representan un proceso secuencial que guía a los investigadores y profesionales en la implementación de modelos de simulación y en la obtención de resultados que contribuyen al entendimiento y la toma de decisiones informadas. A lo largo de la historia, diferentes figuras influyentes han contribuido al avance y la consolidación de las etapas para la co-simulación, aportando conocimientos y metodologías que han revolucionado la forma en que se abordan los sistemas complejos a través de la simulación. En la **Figura 2-9** se puede apreciar cómo se subdividen secuencialmente estas etapas en una co-simulación utilizando la herramienta HELICS, la cual será descrita a detalle en el capítulo siguiente.

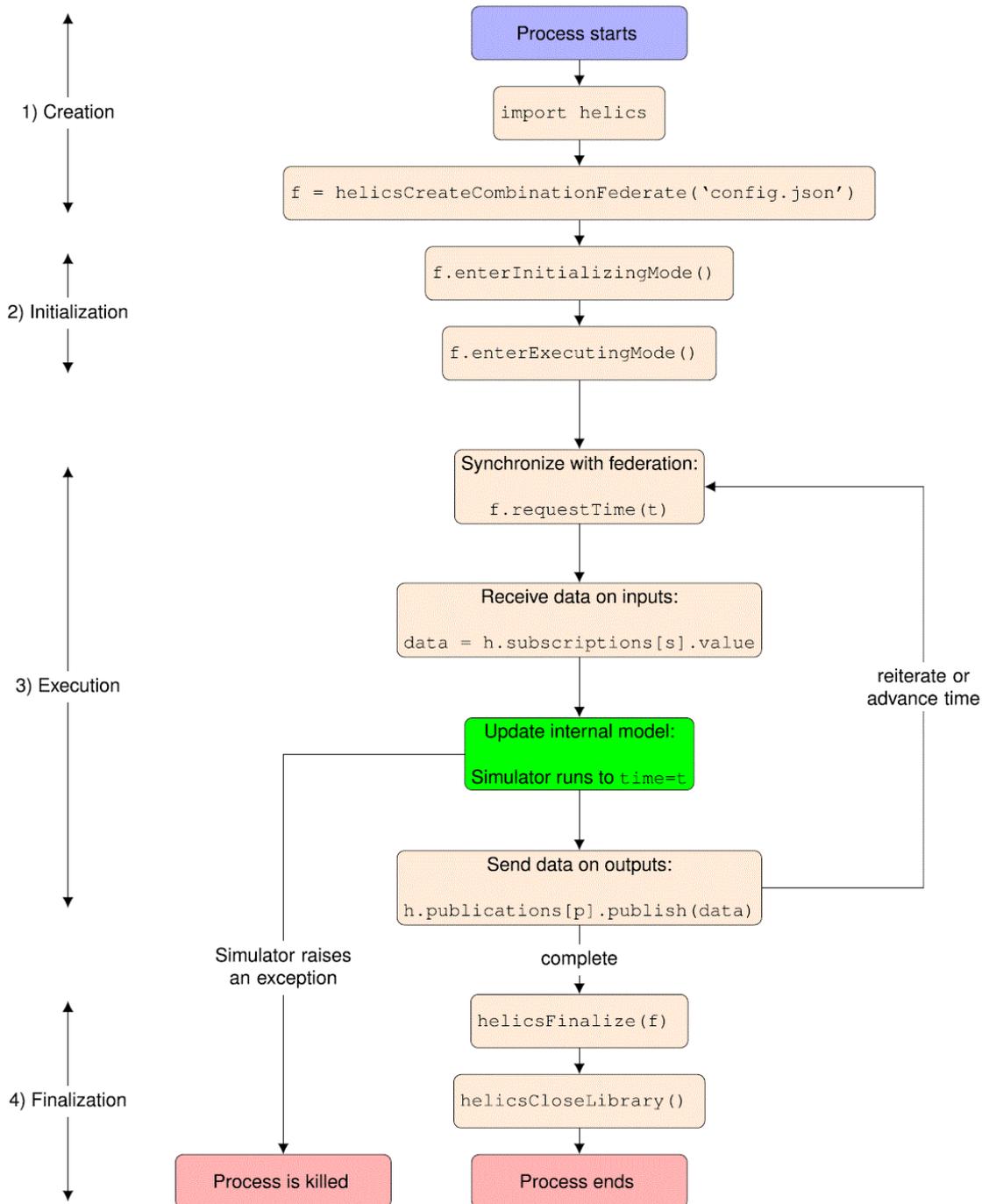
### 2.9.1 Etapa 1: Creación

Durante esta etapa, los investigadores diseñan y desarrollan el modelo que representará el sistema a estudiar. Es esencial establecer claramente los objetivos de la simulación, identificar las variables relevantes y definir las relaciones entre ellas. Además, se selecciona cuidadosamente la metodología y las herramientas de simulación para garantizar la precisión y fiabilidad de los resultados. Figuras influyentes como Claude Shannon y John von Neumann han sentado las bases teóricas para la simulación, lo que ha impulsado el desarrollo de modelos matemáticos y computacionales utilizados en la co-simulación moderna [9].

Durante esta fase, se desarrollan y seleccionan los modelos individuales que representarán los componentes del sistema en la co-simulación. Se establece la arquitectura general de la co-simulación, definiendo cómo interactuarán los modelos entre sí y cómo se compartirá la información. También se identifican los requisitos de hardware y software necesarios para ejecutar la co-simulación. La definición del modelo es un paso crítico en esta etapa, que implica la descomposición del sistema complejo en sus subsistemas individuales y la definición de los límites entre ellos. Se seleccionan los simuladores adecuados para modelar cada subsistema, y se desarrollan los modelos matemáticos o computacionales utilizando dichos simuladores.

### 2.9.2 Etapa 2: Inicialización

La etapa de inicialización en la co-simulación marca el punto de partida para la ejecución del modelo, donde se establecen las condiciones iniciales del sistema y se preparan los recursos esenciales para su funcionamiento. En este contexto, la correcta configuración y calibración del modelo son aspectos críticos para garantizar una representación fiel de la realidad. Investigadores destacados como Jay Forrester y Dennis Meadows han desempeñado un papel significativo en el avance de técnicas de inicialización en la simulación de sistemas dinámicos, aportando herramientas y metodologías que facilitan el arranque adecuado de los modelos y la obtención de resultados precisos y válidos [23, 24].



**Figura 2-9.** Esquema secuencial de las etapas de una co-simulación con HELICS [9].

Durante la fase de inicialización, se llevan a cabo varias actividades fundamentales:

- **Preparación de modelos individuales:** Se establecen las condiciones iniciales y los parámetros de entrada de cada modelo, asegurando su adecuada configuración para la simulación. Además, se configuran los protocolos de comunicación y sincronización entre los modelos para garantizar una interacción fluida durante la ejecución.
- **Carga de datos iniciales:** Se introducen en cada simulador los datos iniciales necesarios para iniciar la co-simulación. Estos datos pueden incluir condiciones iniciales, parámetros de configuración y valores de variables relevantes para el estado inicial de cada subsistema.

- **Validación de datos iniciales:** Se verifica la coherencia y consistencia de los datos iniciales cargados en cada simulador. Este paso es crucial para asegurar que la co-simulación comience desde un estado válido y realista, evitando posibles problemas de inconsistencia que podrían afectar la calidad de los resultados.

### 2.9.3 Etapa 3: Ejecución

El proceso de ejecución en la co-simulación representa el momento crítico en el cual se realiza la simulación propiamente dicha, permitiendo la obtención de resultados que son fundamentales para analizar el comportamiento del sistema en estudio. Durante esta fase, es esencial llevar a cabo un monitoreo exhaustivo del proceso de simulación y validar los resultados obtenidos para asegurar la confiabilidad de las conclusiones derivadas. En la etapa de ejecución, los modelos individuales se ejecutan simultáneamente, interactuando entre sí y respondiendo a los estímulos del entorno de co-simulación. Esta interacción dinámica entre los modelos es esencial para capturar la complejidad y la dinámica del sistema en cuestión. Además, se registra y analiza continuamente el progreso de la co-simulación en tiempo real, lo que permite monitorear el comportamiento del sistema y realizar ajustes si es necesario [8, 25].

Para garantizar una ejecución exitosa de la co-simulación, es crucial establecer una estrategia sólida de sincronización entre los diferentes simuladores. Esto implica iniciar cada simulador con su modelo de subsistema correspondiente y asegurar que el intercambio de datos entre ellos se realice de manera sincronizada y coherente. La correcta implementación de esta estrategia de sincronización es fundamental para mantener la integridad de la simulación y obtener resultados precisos. Además del monitoreo del proceso de simulación, es importante realizar un seguimiento constante del comportamiento de los simuladores individuales para verificar que se estén comportando según lo esperado. En caso de detectar desviaciones o errores, se deben tomar medidas de control adecuadas para ajustar el comportamiento de la co-simulación y garantizar la validez de los resultados obtenidos.

### 2.9.4 Etapa 4: Estado Final

El cierre de una co-simulación es una fase crucial que representa la culminación del proceso de modelado y análisis, donde se examinan y se interpretan los resultados obtenidos para extraer conclusiones significativas y tomar decisiones informadas. Durante esta etapa, es esencial una comunicación efectiva de los hallazgos y recomendaciones derivadas de la simulación, con el propósito de influir en la toma de decisiones y mejorar los sistemas estudiados.

En la fase de estado final de la co-simulación, se establecen criterios de terminación previamente definidos, los cuales pueden basarse en el tiempo de simulación, el logro de objetivos específicos o la detección de errores críticos. Es crucial asegurar que la simulación se detenga de manera adecuada y se recopilen los datos finales de manera ordenada. Se recolectan y consolidan los datos generados durante la co-simulación, representando el comportamiento del sistema en su totalidad. Estos datos se someten a un análisis exhaustivo para extraer información relevante sobre el funcionamiento del sistema, identificando patrones, tendencias y relaciones entre las variables simuladas.

Los resultados de la co-simulación se comparan con datos experimentales o modelos de referencia para validar su precisión y confiabilidad. Este proceso de validación es esencial para garantizar la credibilidad de los resultados obtenidos y su utilidad para la toma de decisiones. Finalmente, los hallazgos y conclusiones derivadas de la co-simulación se documentan exhaustivamente en informes técnicos, presentaciones o publicaciones científicas. La comunicación clara y efectiva de estos resultados es fundamental para compartir el conocimiento generado y contribuir al avance del campo de la co-simulación.

## 2.10 Retos y Limitaciones de la Co-simulación

La co-simulación, si bien ofrece ventajas significativas en el modelado y análisis de sistemas complejos, también presenta varios retos y limitaciones que deben abordarse para su aplicación efectiva. En esta sección se analizan los principales retos asociados a la co-simulación y se exploran posibles estrategias para mitigarlos.

### 2.10.1 Latencia en la Comunicación e Intercambio de Datos

La tarea de Gestión del intercambio de datos maneja la comunicación entre las instancias del modelo, los simuladores y el orquestador mediante la implementación de protocolos de telecomunicaciones que suelen ser la solución más eficaz para esta tarea. En la gestión del intercambio de datos, el principal problema está relacionado con la latencia de comunicación que suele afectar a los protocolos de telecomunicaciones. Más específicamente, la latencia de comunicación en marcos de co-simulación se refiere a la cantidad de tiempo transcurrido desde el envío de las variables de salida de una Instancia de modelo hasta la recepción de la variable como entrada por otra Instancia de modelo. Una latencia elevada puede comprometer el entorno general de co-simulación cuando se trata de restricciones de tiempo estrictas de un simulador en particular que podrían desencadenar internamente un desbordamiento de paso de tiempo. La gestión del intercambio de datos representa uno de los desafíos más importantes para garantizar un marco de co-simulación confiable, preciso y estable [11].

Por ejemplo, en las co-simulaciones distribuidas, donde los componentes de la simulación se distribuyen en múltiples nodos o plataformas computacionales, los consumos generales de comunicación plantean un desafío importante. El intercambio de datos y la sincronización entre componentes distribuidos introduce latencia y sobrecarga, lo que afecta la eficiencia y precisión generales de la co-simulación. Los retrasos en el intercambio de datos entre submodelos pueden alterar el flujo y la precisión de la simulación, de igual forma se produce una escalabilidad limitada, donde las co-simulaciones distribuidas pueden no escalar bien a medida que aumenta el número de subsistemas. Para mitigar esto es necesario emplear protocolos y estrategias de comunicación eficientes. Esto incluye minimizar los requisitos de transferencia de datos, optimizar los mecanismos de paso de mensajes y aprovechar técnicas informáticas paralelas y distribuidas para mejorar la eficiencia de la comunicación. Además, el uso de tecnologías y protocolos de red avanzados puede optimizar aún más la comunicación en co-simulaciones distribuidas.

Otro ejemplo es la compleja regulación del tiempo en el contexto de la simulación híbrida. La simulación híbrida implica la combinación de diferentes paradigmas de simulación, como el tiempo continuo (CT) y los eventos discretos (DE), para modelar sistemas que tienen componentes que evolucionan de manera continua y otros que cambian en momentos discretos. La solución propuesta por AIOMAS (una librería cuyo objetivo principal es facilitar la creación de sistemas distribuidos, como los sistemas multiagente, utilizando Python puro y asyncio [26]) es la capacidad de distribuir la regulación del tiempo y personalizarla según las necesidades específicas del usuario. Esto significa que los usuarios tienen la flexibilidad de implementar la regulación del tiempo de acuerdo con la naturaleza de su sistema y los requisitos de la simulación. Al permitir esta personalización, AIOMAS aborda el desafío de la regulación del tiempo en la co-simulación híbrida, proporcionando a los ingenieros una herramienta más adaptable y efectiva para el modelado de sistemas complejos [11].

### 2.10.2 Problemas de Escalabilidad en Co-simulaciones a Gran Escala

La escalabilidad representa uno de los principales desafíos en la co-simulación, especialmente cuando se trata de sistemas a gran escala. A medida que aumenta el tamaño y la complejidad del sistema, también lo hacen los recursos computacionales necesarios para la co-simulación, lo que puede provocar problemas de escalabilidad y afectar la eficiencia del proceso. Para abordar estos problemas, se requieren enfoques innovadores que optimicen la utilización de recursos, paralelicen los cálculos y agilicen el intercambio de datos

entre los componentes de la co-simulación. Además, es fundamental desarrollar algoritmos y técnicas escalables específicamente diseñadas para co-simulaciones a gran escala, lo que permitirá mejorar el rendimiento y garantizar la viabilidad en aplicaciones del mundo real. Imaginemos una co-simulación masiva que involucra docenas de submodelos de diversas disciplinas [21, 27]. Gestionar tal complejidad puede ser difícil, ya que surgen problemas como:

- **Sobrecarga computacional:** Coordinar numerosas simulaciones simultáneamente puede ejercer presión sobre los recursos informáticos.
- **Gestión de datos:** El intercambio de datos entre un gran número de submodelos puede volverse engorroso y ralentizar la simulación.
- **Complejidad del modelo:** La integración de submodelos altamente detallados puede resultar en una simulación general que sea demasiado costosa computacionalmente.

La buena noticia es que los investigadores y expertos en co-simulaciones están constantemente buscando soluciones para mejorar la escalabilidad. Se desarrollan mecanismos de sincronización más robustos, protocolos de intercambio de datos más eficientes y estrategias computacionales optimizadas para manejar la complejidad de los sistemas a gran escala. De esta forma, las co-simulaciones podrán seguir siendo una herramienta valiosa para estudiar y comprender los sistemas complejos del futuro.

### 2.10.3 Validación y Verificación de Modelos de Co-simulación

Garantizar la precisión y confiabilidad de los modelos de co-simulación es otro desafío crítico. La validación y verificación de modelos de co-simulación implica evaluar la coherencia entre el comportamiento simulado y las observaciones del mundo real, así como la exactitud de las formulaciones matemáticas y los algoritmos computacionales subyacentes. Esto asegura que los modelos estén implementados correctamente y que interactúen según lo esperado dentro del entorno de co-simulación.

La verificación se centra en garantizar la corrección y el funcionamiento adecuado de cada modelo individual dentro del sistema integrado. Esto implica validar que la lógica, las reglas y los algoritmos implementados en cada modelo sean precisos y estén libres de errores. Además, verifica que la implementación de cada modelo sea correcta según sus especificaciones y requisitos, validando que represente con precisión el comportamiento del sistema que se pretende simular. Asimismo, asegura que las interacciones entre los modelos ocurran como se espera dentro del entorno de co-simulación, validando los protocolos de comunicación, los mecanismos de intercambio de datos, los procesos de sincronización y la integración general de los modelos individuales.

Por otro lado, la validación en la co-simulación se enfoca en evaluar la capacidad colectiva de los modelos integrados para replicar con precisión el comportamiento del sistema real. Esto implica asegurar que los modelos integrados, al trabajar juntos en el entorno de co-simulación, reproduzcan con precisión el comportamiento del sistema real. Además, la validación garantiza que la configuración de co-simulación produzca resultados confiables que puedan utilizarse para la toma de decisiones y el análisis. La validación en la co-simulación también sirve como un mecanismo de garantía de calidad para verificar que los modelos integrados funcionen armoniosamente juntos para proporcionar una representación integral del sistema real. En resumen, tanto la verificación como la validación son procesos esenciales en la co-simulación para garantizar que los modelos integrados representen con precisión el sistema real y produzcan resultados confiables que puedan utilizarse para la toma de decisiones y el análisis.

### 2.10.4 Otros Desafíos de la Co-simulación

Abordar los desafíos esenciales de la co-simulación es fundamental para garantizar su eficacia y confiabilidad en tareas de modelado y análisis de sistemas complejos. La co-simulación presenta otros desafíos, profundicemos en cada desafío y analicemos estrategias implícitas para mitigarlos.

1. **Desafíos matemáticos:** La naturaleza de los simuladores y la rigidez de las ecuaciones subyacentes en la co-simulación pueden generar garantías de convergencia débiles. El intercambio de información entre subproblemas normalmente ocurre solo después de la convergencia, lo que plantea desafíos para garantizar una comunicación precisa y oportuna entre los componentes de la simulación. El empleo de solucionadores iterativos y métodos de aceleración de confluencia puede mejorar la solidez y eficacia de los algoritmos de co-simulación [28]. Además, la integración de formas de reducción de modelos y estilos de análisis de receptividad puede mejorar las garantías de confluencia y acelerar la confluencia de la simulación.
2. **Complejidad de la interfaz:** Integrar diferentes herramientas y modelos de simulación de diversos dominios en un marco de co-simulación puede resultar complejo. Simplificar la complejidad de la interfaz implica estandarizar los formatos de datos, los protocolos de comunicación y los mecanismos de sincronización en diferentes herramientas y modelos de simulación [28]. Adoptar interfaces estándar de la industria, como *Functional Mock-up Interface (FMI)*, puede facilitar la integración y la interoperabilidad entre componentes de simulación diversos. Además, desarrollar adaptadores especializados y capas de middleware puede agilizar los procesos de interfaz e intercambio de datos en marcos de co-simulación.
3. **Requisitos de recursos:** La co-simulación a menudo requiere importantes recursos computacionales para ejecutar múltiples simuladores simultáneamente y manejar el intercambio de información entre ellos. Gestionar los requisitos de recursos implica optimizar los recursos computacionales e infraestructura para una ejecución eficiente de co-simulación. Emplear técnicas de computación paralela, como el multiprocesamiento o la computación distribuida, puede aprovechar el paralelismo para mejorar el rendimiento y la escalabilidad [28]. Además, utilizar recursos de computación en la nube y clústeres de computación de alto rendimiento (HPC) puede proporcionar escalabilidad y aprovisionamiento de recursos bajo demanda para estudios de co-simulación a gran escala.

Al abordar estos desafíos a través de algoritmos avanzados, protocolos de comunicación eficientes y prácticas sólidas de desarrollo de software, los investigadores pueden mejorar la efectividad y confiabilidad de la co-simulación para tareas de modelado y análisis de sistemas complejos.

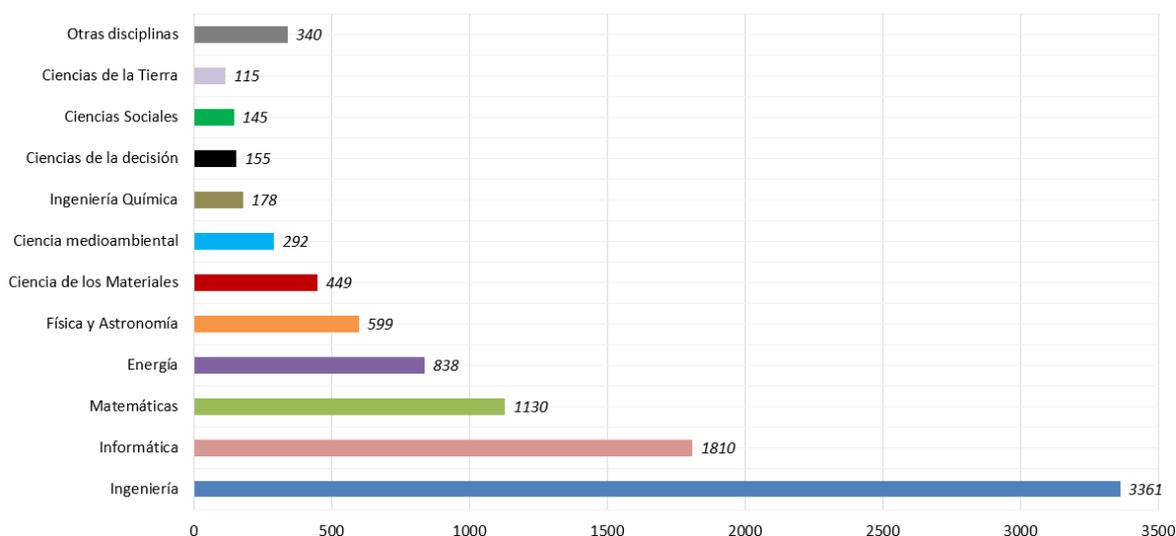
## 2.11 Importancia de la Co-simulación

En los últimos años se ha podido resaltar como la co-simulación desempeña un papel crucial, no solo en la ingeniería actual, sino en múltiples áreas de investigación. Esto debido a que permite el análisis detallado y la optimización de sistemas complejos. Al integrar modelos de diferentes componentes y disciplinas, la co-simulación proporciona una visión más completa del comportamiento del sistema, lo que facilita la toma de decisiones informada, la mejora del rendimiento y la eficiencia de los sistemas avanzados. La co-simulación ofrece una serie de ventajas sobre otras formas de combinación de modelos, como la construcción de un modelo completo del sistema en una sola herramienta o el intercambio de modelos entre herramientas. Estas ventajas incluyen:

- Su modularidad y alta flexibilidad, que permite construir simulaciones de sistemas complejos a partir de la incorporación de varios simuladores independientes, donde cada uno está dedicado a modelar subsistemas específicos. Esto permite un análisis integral de los sistemas interconectados, proporcionando una visión más holística del sistema complejo.
- La capacidad de incluir *Hardware-in-the-Loop (HIL)*, *Software-in-the-Loop (SIL)* y *Human-in-the-Loop (HIL)* con relativa facilidad y consistencia amplía significativamente su utilidad y aplicabilidad en el desarrollo y validación de sistemas complejos.

- La capacidad de aprovechar cadenas de herramientas especializadas y conocimientos específicos del dominio ya utilizados para una curva de aprendizaje reducida. Al aprovechar formatos de datos estandarizados los usuarios con experiencia en simuladores individuales pueden configurar y ejecutar co-simulaciones fácilmente sin una capacitación extensa.
- La naturaleza abierta de la co-simulación, junto con un enfoque de caja negra que protege los derechos de propiedad intelectual (IPR). También los usuarios pueden configurar y definir parámetros de simulación, y adaptar el software para abordar necesidades operativas o de investigación específicas.
- La eficiencia en el desarrollo de modelos, ya que permite el desarrollo paralelo de modelos y su reutilización. Asimismo, la ejecución en paralelo de simuladores, con el potencial de acelerar las simulaciones de sistemas completos [29].

Estas ventajas han llevado a los investigadores a utilizar la co-simulación en distintos ámbitos y han dado lugar a un aumento progresivo del número de publicaciones en el campo de la co-simulación para diversos ámbitos en los últimos años, como se muestra en la **Figura 2-10**. Este gráfico se realizó con datos de la base de datos bibliográficos y herramienta de análisis de citas desarrollada por *Elsevier* llamada ‘Scopus’, se examinaron los proyectos relacionados con la palabra clave "co-simulation" entre 2020 y 2024. Como se puede ver que la mayoría de las publicaciones se pueden asignar a los campos de la Ingeniería (35.71%), seguido de la Informática (19.23%) y las Matemáticas (12.01%).



**Figura 2-10.** Publicaciones por área de investigación de ‘co-simulación. Elaboración propia.

Sin embargo, debido a la complejidad del proceso de co-simulación, puede verse afectado por factores como el uso de software específico del ámbito, la elección de lenguajes de programación, los solucionadores de co-simulación, otras herramientas de apoyo y la interoperabilidad de estas herramientas. Por lo general, estos retos se abordan mediante colaboraciones dentro de la comunidad investigadora. Por lo tanto, la mayoría de las implementaciones encontradas en la literatura se basan en colaboraciones de diferentes países. En la **Figura 2-11** se puede apreciar los 10 países con más trabajos sobre co-simulación realizados entre 2020 y 2024 alrededor del mundo.

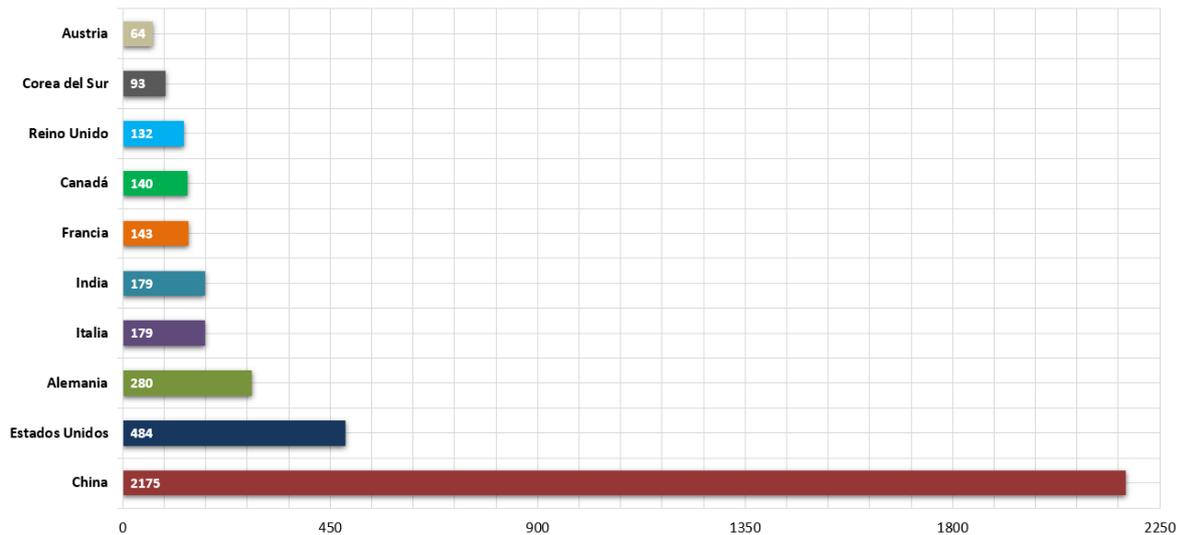


Figura 2-11. Países con mayor número de publicaciones sobre 'co-simulación'. Elaboración propia.

### 2.11.1 Aplicaciones de la Co-simulación en la Actualidad

A continuación, se indican de forma resumida algunos sectores y campos clave (fuera de la ingeniería eléctrica) en los que la co-simulación se aplica:

- 1. Industria automovilística:** La co-simulación se utiliza ampliamente en la ingeniería de automoción para la dinámica de vehículos, el análisis de la cadena cinemática y el desarrollo de sistemas avanzados de asistencia al conductor (ADAS). Por ejemplo, en el desarrollo de vehículos eléctricos (VE), la co-simulación permite integrar simulaciones mecánicas, eléctricas y de sistemas de control para optimizar la eficiencia energética y el rendimiento, al tiempo que se garantiza la seguridad y la fiabilidad.
- 2. Aeroespacial y Defensa:** En las aplicaciones aeroespaciales, la co-simulación se emplea para el diseño de aeronaves, la dinámica de vuelo y la planificación de misiones. La interacción en tiempo real entre simulaciones aerodinámicas, estructurales y de sistemas de control facilita la predicción precisa del rendimiento y mejora la optimización del diseño. Por ejemplo, la co-simulación se utiliza en el diseño y las pruebas de vehículos aéreos no tripulados (UAV) para garantizar la estabilidad y el control en diversas condiciones de vuelo.
- 3. Sistemas de calefacción, ventilación y aire acondicionado:** En la gestión energética de edificios y el diseño de sistemas de calefacción, ventilación y aire acondicionado (HVAC), la co-simulación permite optimizar la eficiencia energética, el confort interior y el rendimiento térmico. La integración de herramientas de simulación energética de edificios con modelos de sistemas HVAC permite evaluar estrategias de ahorro energético, como técnicas de diseño pasivo y algoritmos de control avanzados.
- 4. Ingeniería de procesos químicos:** La co-simulación se utiliza en ingeniería de procesos químicos para el diseño, optimización y control de procesos químicos como reactores, columnas de destilación e intercambiadores de calor. Al integrar software de simulación de procesos con modelos de sistemas de control, la simulación conjunta permite analizar la dinámica de los procesos, el consumo de energía y la calidad de los productos.
- 5. Ingeniería sanitaria y biomédica:** En ingeniería sanitaria y biomédica, la co-simulación se emplea para modelar sistemas fisiológicos, dispositivos médicos y sistemas de administración de fármacos. Al combinar modelos computacionales de procesos biológicos con simulaciones de dispositivos, la co-simulación facilita el diseño y ensayo de intervenciones médicas y estrategias de tratamiento. Algunos

ejemplos son la simulación de la cinética de liberación de fármacos a partir de dispositivos implantables y la optimización de algoritmos de control para robots médicos utilizados en procedimientos quirúrgicos.

6. **Ciencias medioambientales y control de la contaminación:** La co-simulación se aplica en la ciencia medioambiental y el control de la contaminación para evaluar el impacto de las actividades humanas en los sistemas naturales y desarrollar estrategias de mitigación. Al combinar modelos de dispersión atmosférica, procesos hidrológicos y transporte de contaminantes, la simulación conjunta permite realizar estudios exhaustivos sobre la calidad del aire y del agua, la salud de los ecosistemas y la sostenibilidad medioambiental. Algunos ejemplos reales son la simulación de la dispersión de contaminantes procedentes de instalaciones industriales y la evaluación de técnicas de remediación de lugares contaminados.
7. **Servicios financieros y gestión de riesgos:** La co-simulación encuentra aplicación en los servicios financieros y la gestión de riesgos para modelizar sistemas financieros complejos, optimizar carteras y evaluar riesgos. Al integrar modelos económicos con datos de mercado y herramientas de análisis de riesgos, la co-simulación permite realizar análisis de escenarios, pruebas de resistencia y apoyo a la toma de decisiones para estrategias de inversión y mitigación de riesgos. Algunas de las aplicaciones incluyen la simulación de riesgos sistémicos en los mercados financieros y la evaluación de estrategias de cobertura para gestionar la exposición al riesgo de las carteras.

Estos ejemplos son solo una pequeña muestra donde se demuestra la versatilidad de la co-simulación en un amplio abanico de sectores y campos, y ponen de manifiesto su eficacia a la hora de abordar complejos retos de ingeniería, científicos y socioeconómicos.

### 2.11.2 La Co-simulación como Herramienta de Colaboración Interdisciplinaria

La co-simulación interdisciplinaria emerge como una herramienta poderosa para mejorar la modelización y análisis de sistemas multi-energéticos. Al aprovechar la diversidad de perspectivas, la integración de conocimientos y la promoción de la innovación, la co-simulación interdisciplinaria ofrece una vía prometedora para abordar los desafíos de la transición hacia sistemas energéticos más sostenibles y eficientes. Futuras investigaciones podrían explorar aún más los métodos y enfoques específicos para la implementación efectiva de la co-simulación interdisciplinaria en este campo en constante evolución.

Es importante destacar que la *colaboración interdisciplinaria* es un enfoque más amplio que abarca la colaboración entre diferentes disciplinas en cualquier aspecto de la investigación o desarrollo, mientras que la *co-simulación interdisciplinaria* se centra específicamente en la integración y la simulación conjunta de modelos de diferentes disciplinas para analizar sistemas complejos. Sin embargo, es importante destacar que la co-simulación interdisciplinaria puede ser una forma específica de colaboración interdisciplinaria en el contexto de proyectos de investigación y desarrollo que requieren análisis detallado de sistemas complejos.

### 2.11.3 La Co-simulación en el ámbito de las Redes de Distribución

A medida que más y más usuarios de diversas magnitudes se conectan a la red, se vuelve más compleja y desafiante de gestionar. En el caso de las redes de distribución, la co-simulación puede ser utilizada para estudiar la interacción entre diferentes áreas de la red, como líneas de transmisión en transporte y redes de baja tensión en distribución. A medida que la red eléctrica se expande con la conexión de un número creciente de usuarios, su gestión se torna más compleja y desafiante. En este contexto, la co-simulación emerge como una herramienta clave. En lugar de abordar la red en su totalidad como un rompecabezas gigante, la co-simulación permite descomponer este desafío en piezas más pequeñas y manejables. Esta técnica posibilita la simulación simultánea del comportamiento de múltiples sistemas físicos, siendo especialmente relevante en el caso de las redes de distribución.

La adopción de co-simulaciones presenta ventajas fundamentales en el ámbito de los sistemas de distribución eléctrica. Estas ventajas incluyen una visión más precisa y completa al combinar simuladores especializados, lo que conduce a resultados más confiables; la capacidad de optimizar la eficiencia y tomar decisiones informadas al observar el sistema en su conjunto; la exploración de escenarios complejos y realistas para la investigación avanzada; la detección y resolución anticipada de problemas mediante la simulación de situaciones reales, mejorando la confiabilidad del sistema; y la promoción de la colaboración entre diferentes expertos al permitirles contribuir con sus conocimientos en una única plataforma, fomentando la interdisciplinariedad y el intercambio de ideas.

# 3 ENTORNOS DE CO-SIMULACIÓN CON ENFOQUE EN SISTEMAS ELÉCTRICOS

---

*“La lucha a lo largo del camino solo pretendía dar forma a tu propósito”.*

*- Chadwick Boseman -*

La creciente complejidad de los sistemas eléctricos actuales pone a prueba las prácticas tradicionales de modelado y simulación. Una combinación de economía, electrónica de potencia, sistemas de control, redes de comunicaciones, participación de la demanda y big data son ahora actores que contribuyen al funcionamiento de los sistemas eléctricos actuales. En el pasado, estos agentes se han gestado de forma individual y separada, pero ahora se comunican cada vez más entre sí y estas interacciones exigen un medio para reunirlos y analizarlos con el fin de obtener una imagen completa del sistema. Aunque estos cambios plantean nuevos retos, también ofrecen nuevas oportunidades para mejorar el rendimiento de los sistemas eléctricos, sobre todo si se combinan con los recursos informáticos cada vez mayores de que disponen los ingenieros de planificación [4].

## 3.1 Estado Actual de los Entornos de Co-simulación

Los entornos, herramientas y plataformas de co-simulación han experimentado un crecimiento significativo en el campo de la ingeniería eléctrica, abordando la necesidad de integrar diferentes sistemas y componentes para el análisis integral de sistemas de energía eléctrica. En los últimos años la comunidad de sistemas eléctricos ha desarrollado o explorado muchas plataformas de co-simulación, que se encuentran en distintos niveles de madurez.

### 3.1.1 Principales áreas de Investigación de los Entornos de Co-simulación

Hasta el momento, la investigación mediante el uso de entornos o plataformas de co-simulación en el ámbito de los sistemas eléctricos ha abarcado una gran variedad de aspectos. Un estudio similar realizado en 2018 demostró que se había utilizado la co-simulación como herramienta de investigación en 30 diferentes ramas de investigación del ámbito de sistemas eléctricos en Europa [12]. A continuación, se exploran las principales áreas de investigación en las que se ha utilizado la co-simulación en los últimos años [10, 30], incluyendo aquellas identificadas en estudios previos:

1. **Recursos Energéticos Distribuidos (DER):** Este tipo de investigación abarca desde el almacenamiento de energía y las unidades de generación independientes de las compañías de servicios públicos, hasta el desarrollo de nuevas capacidades de almacenamiento. Este enfoque incluye la integración de fuentes de energía renovables como unidades eólicas o fotovoltaicas para proporcionar

energía adicional a la red y hacerla más estable, especialmente durante los momentos de máxima demanda. Además, se exploran tecnologías de almacenamiento de energía en la red para mejorar la flexibilidad y la eficiencia del sistema eléctrico. Las co-simulaciones han sido utilizadas para estudiar la integración de paneles fotovoltaicos para el consumo de energía en hogares, estrategias óptimas para la recarga de vehículos eléctricos, y las estrategias de conexión de vehículos eléctricos para estabilizar el voltaje de la red. Ejemplos: [31, 32, 33].

2. **Almacenamiento y Movilidad Eléctrica:** Este tipo de investigación se enfoca principalmente en el estudio de los vehículos eléctricos y las oportunidades que brinda su uso como unidades móviles de almacenamiento de energía. Otros aspectos significativos de investigación son la integración de la red cableada e inalámbrica, los bancos de baterías, la integración a gran escala en la red eléctrica y las estaciones de carga. Las co-simulaciones han sido fundamentales en este campo, permitiendo la evaluación de nuevos algoritmos de carga para vehículos eléctricos, así como la simulación de eventos de carga en la red eléctrica. Esto incluye el análisis de tecnologías inalámbricas y de sensores para la carga de vehículos eléctricos, así como la investigación de algoritmos de carga flexibles para optimizar la distribución de la carga en la red. Ejemplos: [34, 35, 36].
3. **Fiabilidad y Control de Área Amplia:** Este tipo de investigación se centra en diagnosticar, anticipar y responder a problemas antes de que ocurran interrupciones en los sistemas de energía eléctrica. En el contexto de las redes inteligentes, la propiedad de autocuración implica que estas redes pueden detectar problemas y reanudar operaciones normales de manera automática, reduciendo significativamente el tiempo necesario para reparaciones. Para lograr esto, se requiere un esfuerzo importante en fortalecer y asegurar la protección, el control y la red de comunicaciones. Las co-simulaciones han sido utilizadas frecuentemente para evaluar el rendimiento y la confiabilidad de los sistemas de monitoreo y control a gran escala en las redes eléctricas. Estas co-simulaciones analizan los efectos combinados de comunicación, red eléctrica y niveles de potencia para el monitoreo, protección y control a nivel amplio. Se utilizan datos proporcionados por unidades de medición de fase para analizar sistemas de protección y control a gran escala, con el objetivo de prevenir fallas y mejorar la eficiencia operativa de las redes eléctricas. Ejemplos: [37, 38, 39].
4. **Eficiencia Energética del Consumidor:** Este es un objetivo clave para reducir el consumo de energía durante períodos de alta demanda o cuando la fiabilidad del suministro de energía está en riesgo. Este efecto es crucial para optimizar el equilibrio de la energía distribuida en la red eléctrica. Históricamente, las investigaciones en eficiencia energética del consumidor han sido un área de enfoque en estudios de co-simulación, y seguirá siendo relevante en el futuro. Las co-simulaciones se han utilizado para el control de carga, modelar el comportamiento de clientes y prosumidores, control de voltaje para el equilibrio de carga entre la demanda y la oferta, simulación de cargas residenciales, estudios de demanda y respuesta, y fijación de precios de energía. Estas simulaciones permiten comprender mejor el comportamiento de los consumidores y prosumidores, así como diseñar estrategias efectivas para optimizar el uso de energía y mantener el equilibrio en la red eléctrica. Ejemplos: [40, 41, 42].
5. **Gestión de la Red de Distribución:** Este tipo de gestión se centra en maximizar el rendimiento de los alimentadores, transformadores y otros componentes de los sistemas de distribución en red, e integrarlos con los sistemas de transmisión y las operaciones de los clientes. Las co-simulaciones se han enfocado en mecanismos de sincronización para sistemas SCADA, equilibrio de carga y control de alimentadores de energía, regulación de voltaje en redes de distribución de energía y regulación de unidades de medición de fasores. Estos estudios han analizado cómo mejorar la coordinación y el control de los sistemas de distribución de energía eléctrica, lo que permite una operación más eficiente y confiable de la red eléctrica en su conjunto. Ejemplos: [43, 44, 45].
6. **Infraestructura Avanzada de Medición:** Este tipo de investigación se centra en la tecnología de

medidores inteligentes para monitorear y detectar diferentes tipos de distorsión en la calidad de la energía eléctrica. Un medidor inteligente es un dispositivo electrónico que registra el consumo de energía eléctrica y comunica la información al proveedor de electricidad para monitoreo y facturación. El objetivo principal de esta área de investigación es la integración de diversas tecnologías que proporcionen una conexión inteligente entre los consumidores y los operadores del sistema. Las co-simulaciones se han utilizado para modelar escenarios de demanda y respuesta y precios de energía, estudiando tanto la influencia de los aspectos físicos de la red como del mercado energético. Estas simulaciones proporcionan información valiosa sobre cómo los cambios en la infraestructura de medición pueden afectar tanto el funcionamiento físico de la red como las decisiones en el mercado energético, lo que permite a los investigadores y operadores de sistema tomar decisiones informadas para mejorar la eficiencia y la confiabilidad del sistema eléctrico. Ejemplos: [8, 46, 47].

- 7. Ciberseguridad y Comunicaciones de Red:** Las redes eléctricas inteligentes se construyen sobre infraestructuras de tecnologías de la información y comunicación, lo que las hace vulnerables a amenazas de ciberataques. La ciberseguridad en este contexto considera protocolos de comunicación específicos en diversos dominios, mientras que la fiabilidad del flujo de información dentro de la infraestructura de la SG es fundamental para la corrección de las operaciones. Las co-simulaciones han investigado la integración de simulaciones de fallas de red, ataques de inyección de datos en diferentes partes de la infraestructura de las redes inteligentes, así como la evaluación de algoritmos de detección de intrusiones y la identificación de vulnerabilidades en la integración de la red eléctrica y las tecnologías de la información y comunicación. También se han desplegado co-simulaciones para analizar el rendimiento de las redes de comunicación en la integración dentro de la infraestructura de las redes inteligentes, evaluando las redes de comunicación inalámbricas y cableadas, así como la comunicación entre los componentes de la red y los sistemas de control. Ejemplos: [48, 49, 50].

Estas son solo algunas de las principales áreas de investigación en las cuales se han implementado co-simulaciones en el ámbito de los sistemas eléctricos. Con esto se puede destacar la capacidad de la co-simulación para evaluar el rendimiento de sistemas complejos y explorar diversas estrategias de manera virtual ofrece un camino hacia la creación de redes eléctricas más eficientes, confiables y sostenibles para el futuro.

## 3.2 Principales Plataformas de Co-simulación Eléctrica

Como ya se ha explicado en secciones anteriores, las herramientas o plataformas de co-simulación desempeñan un papel crucial en el análisis, diseño y desarrollo de sistemas complejos, ya que permiten a los ingenieros predecir el comportamiento de estos sistemas antes de que se construyan. En la actualidad existen varias herramientas o plataformas de co-simulación en el mercado, cada una con sus propias características y capacidades. A continuación, se presenta algunos de los programas de co-simulación más conocidos en el ámbito de los sistemas eléctricos.

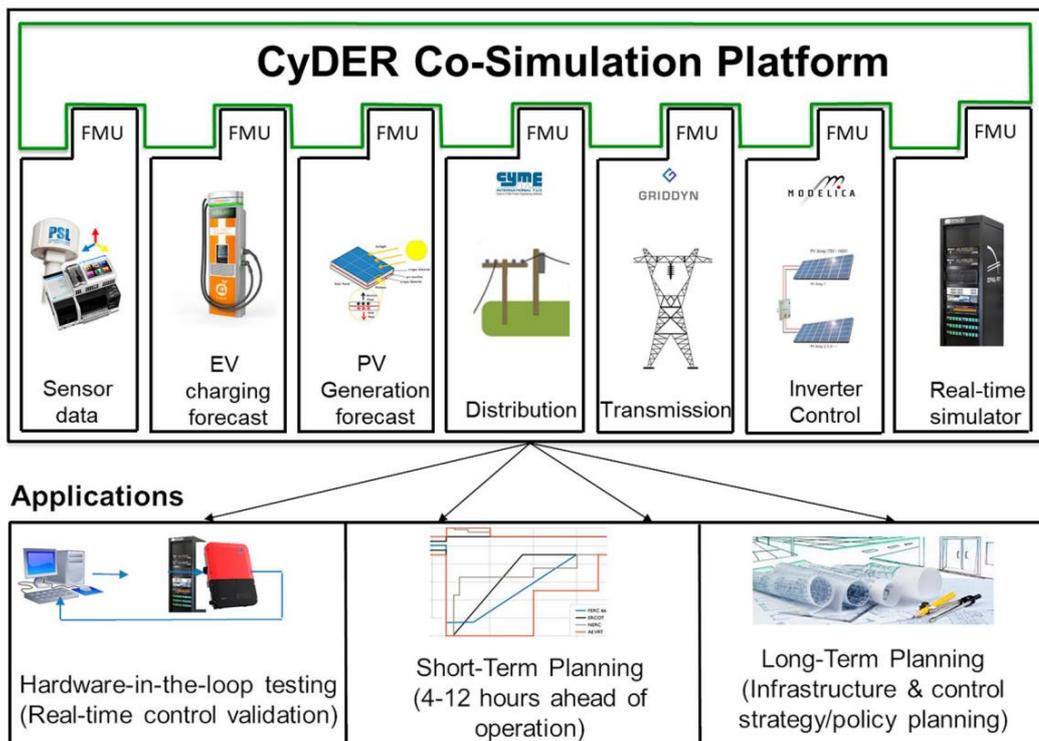
### 3.2.1 CyDER

CyDER (*Cyber-Physical Co-simulation Platform for Distributed Energy Resources in Smart Grids*) es una plataforma de co-simulación de código abierto, modular y escalable diseñada para redes eléctricas con importantes recursos energéticos distribuidos (DER). Desarrollada en colaboración por socios como el Laboratorio Nacional Lawrence Berkeley (LBNL), el Laboratorio Nacional Lawrence Livermore (LLNL), PG&E, SolarCity y ChargePoint. CyDER aborda los retos derivados de la creciente integración de los DER, como las interacciones dinámicas entre las redes de transmisión y distribución, y la mayor complejidad del modelado. Es una plataforma de co-simulación basada en el estándar FMI, que permite coordinar el intercambio de información entre diversos componentes. A diferencia de los enfoques tradicionales que replican todos los

modelos dentro de un único programa de simulación, CyDER acopla simuladores y modelos de terceros a través de *Functional Mock-up Units* (FMU) conformes con el estándar FMI, lo que facilita simulaciones holísticas al tiempo que aprovecha los estándares industriales.

Las capacidades de CyDER se extienden a las simulaciones *Hardware-in-the-Loop* (HIL), acoplando componentes de software con dispositivos de hardware como sistemas fotovoltaicos y baterías. La plataforma facilita el análisis de la capacidad de alojamiento fotovoltaico y admite controles de inversores inteligentes, almacenamiento en baterías y simulaciones de carga de vehículos eléctricos. En concreto, CyDER ofrece una configuración experimental que incluye inversores fotovoltaicos a escala residencial, almacenamiento en baterías, un simulador de red en tiempo real y micro unidades de medición fasorial (PMU), todo ello interconectado mediante el estándar FMI para la simulación conjunta.

En términos de desarrollo técnico, CyDER implica el diseño de especificaciones e interfaces FMI entre módulos, la recopilación de datos para la integración en red y fotovoltaica, y el desarrollo de soluciones para simulaciones de series temporales cuasiestáticas (QSTS). La plataforma utiliza PyFMI como algoritmo maestro para vincular, intercambiar y sincronizar datos entre las FMU en tiempo de ejecución, proporcionando una interfaz fácil de usar para simular sistemas acoplados.



**Figura 3-1.** Plataforma de co-simulación CyDER con aplicaciones específicas [51].

### 3.2.2 DACCOSIM-NG

DACCOSIM-NG (*Distributed Architecture for Controlled CO-Simulation, New Generation*) es un sofisticado entorno de co-simulación desarrollado por EDF y CentraleSupélec, diseñado para agilizar y simplificar las co-simulaciones, en particular para sistemas energéticos, permitiendo una cooperación sin fisuras entre varios simuladores conocidos como *Functional Mockup Units* (FMU). Proporciona un marco gráfico para diseñar gráficos de cálculo y configurar automáticamente el algoritmo maestro asociado, facilitando a los usuarios la configuración y ejecución de co-simulaciones.

Implementado en Java y utilizando la caja de herramientas JavaFMI para envolver FMUs, este entorno soporta la integración de simuladores de diferentes fuentes importándolos como FMUs. Esto permite que las

simulaciones desarrolladas en diferentes lenguajes de programación y desplegadas en varias plataformas se integren sin problemas en el entorno de co-simulación. La arquitectura de DACCOSIM-NG incluye un algoritmo maestro jerárquico que facilita la sincronización descentralizada y el intercambio de datos entre las FMU. Esta arquitectura permite la ejecución paralela de las FMU en múltiples nodos de computación, aprovechando los nodos de clúster de computación multinúcleo para soportar co-simulaciones a gran escala. El algoritmo maestro implementa un mecanismo distribuido de negociación del tamaño del paso para garantizar la sincronización entre las FMU.

Este entorno ofrece tanto una interfaz gráfica de usuario (GUI) como una interfaz de línea de comandos (CLI) para mayor comodidad del usuario, facilitando el diseño de los gráficos de co-simulación, la configuración de los casos de simulación y la implementación del gráfico en las tareas maestras. Además, proporciona un amplio conjunto de componentes para definir gráficos de co-simulación, los cuales permiten a los usuarios especificar cómo se intercambian las variables entre los nodos dentro del gráfico de co-simulación, permitiendo la simulación de sistemas de tiempo continuo con pasos de tiempo constantes o variables.

El paquete de computación de DACCOSIM-NG controla todos los aspectos del proceso de co-simulación, incluyendo la ejecución de tareas, el control de los pasos temporales, la detección de eventos, la comunicación inter-FMU y la gestión de la simulación. Utiliza arquitecturas de ejecución paralelas y distribuidas para maximizar el rendimiento y la escalabilidad, distribuyendo las FMU entre los nodos del clúster e implementando un maestro de control jerárquico para gestionar los cálculos concurrentes.

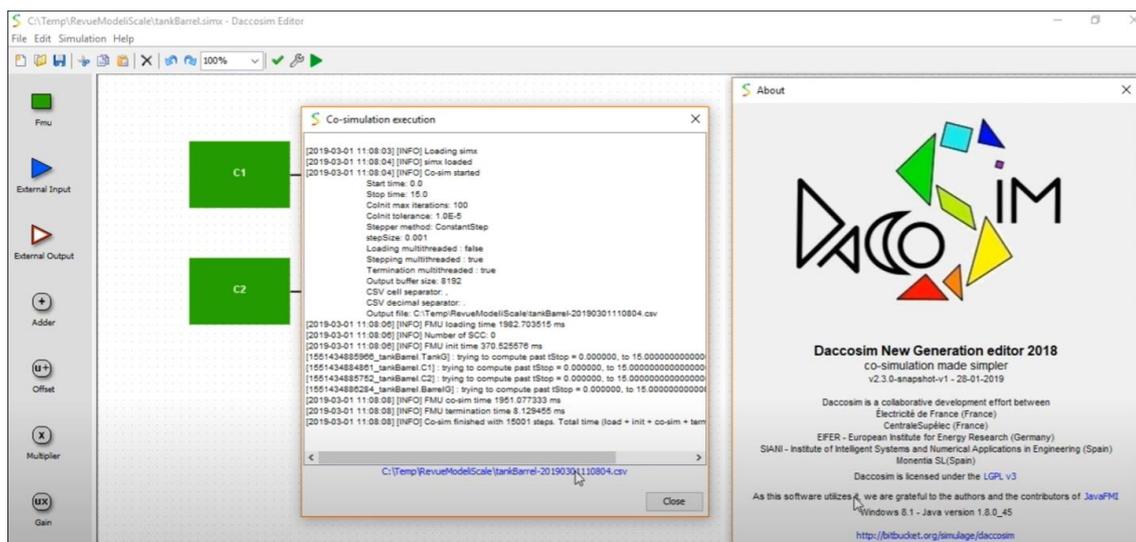


Figura 3-2. Interfaz grafica de la plataforma de co-simulación DACCOSIM-NG [52].

### 3.2.3 EPOCHS

EPOCHS (*Electric Power and Communication Synchronizing Simulator*) es un completo sistema de co-simulación diseñado para enlazar múltiples simuladores con el fin de investigar escenarios electromagnéticos y electromecánicos relacionados con las redes de comunicaciones. Integra a la perfección tres simuladores principales: *PSCAD/EMTDC* para la simulación de transitorios electromagnéticos, *PSLF* para la simulación de transitorios electromecánicos y *NS2* para la simulación de redes de comunicación.

Una de sus características más notables es su capacidad para utilizar simuladores comerciales y de investigación de alta calidad sin modificar sus motores principales. Esto se consigue mediante el uso de componentes middleware como la Infraestructura en Tiempo de Ejecución (RTI), que facilita el intercambio de datos y mantiene la sincronización temporal entre todos los componentes de simulación. EPOCHS introduce otro componente que es el *AgentHQ*, que sirve de entorno unificado para los agentes simulados, facilitando la interacción con otros componentes de EPOCHS.

La sincronización entre los simuladores sigue un modelo escalonado en el tiempo, en el que el tiempo de simulación se divide en intervalos de longitud fija. En cada punto de sincronización se produce un intercambio de datos entre los simuladores, lo que permite una coordinación adecuada de los eventos de simulación. Sin embargo, surgen dificultades debido a la distinta naturaleza de los simuladores, especialmente entre simuladores de tiempo continuo como PSLF/EMTDC y simuladores de eventos discretos como NS2. Para hacer frente a estos desafíos, EPOCHS implementa estrategias para gestionar los errores introducidos por eventos asíncronos, como los eventos de mensajes entre puntos de sincronización. Se presta especial atención a la selección de los intervalos de sincronización para minimizar los errores acumulados en el sistema, al tiempo que se garantiza una ejecución eficiente de la simulación.

### 3.2.4 FNCS

'*Framework for Network Co-Simulation*' (FNCS) es una potente herramienta de co-simulación que permite integrar simuladores de redes eléctricas y de comunicaciones mediante un enfoque federado. A diferencia de los métodos tradicionales de co-simulación, FNCS funciona ejecutando cada simulador en su propio proceso, al tiempo que gestiona la sincronización y el envío de mensajes entre ellos. Este enfoque permite la co-simulación sin fisuras de simuladores de redes eléctricas de transmisión y distribución junto con simuladores de redes de comunicaciones. Implementado principalmente en C++, FNCS ofrece interfaces para otros lenguajes como C, Java, Python y MATLAB, lo que aumenta su versatilidad y usabilidad en distintos entornos. FNCS emplea un esquema de sincronización en dos fases para garantizar la entrega puntual y coherente de mensajes entre simuladores, superando los retos asociados a la sincronización temporal.

La arquitectura de FNCS consta de varios componentes clave, como el corredor FNCS y el comunicador entre simulaciones, que facilitan el control centralizado de los procesos de los simuladores y gestionan la comunicación entre el corredor y los simuladores. Además, el componente de gestión de la comunicación proporciona interfaces para enviar y recibir mensajes, lo que permite una comunicación fluida entre los nodos de la simulación.

FNCS ofrece tres estrategias de sincronización -conservadora, especulativa de conmutación por error y especulativa de recálculo-, cada una de ellas adaptada a escenarios de simulación específicos y a limitaciones de recursos. Estas estrategias garantizan la entrega puntual y coherente de los mensajes, al tiempo que se adaptan a las distintas características de rendimiento y disponibilidad de recursos.

A pesar de sus características y capacidades innovadoras, FNCS ha sido objeto de críticas por su escalabilidad y la falta de capacidades clave como la 'co-iteración' dentro de un paso de tiempo para garantizar la convergencia entre federados. No obstante, FNCS representa una dirección prometedora en el campo de la co-simulación, especialmente para el estudio de infraestructuras de redes inteligentes y sistemas ciberfísicos.

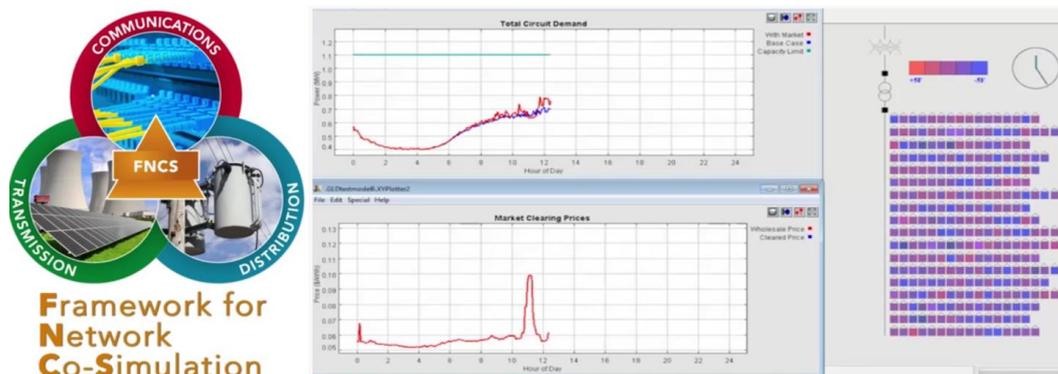


Figura 3-3. Logo y ejemplo de la interfaz de la plataforma de co-simulación FNCS [53].

### 3.2.5 GECO

GECO (*Global Event-driven CO-simulation platform*) es una sofisticada herramienta diseñada para modelar y simular esquemas de supervisión, protección y control de sistemas eléctricos de área extensa. Con ella se aborda el reto de construir sistemas de emulación de hardware para sistemas eléctricos a gran escala, que pueden resultar prohibitivamente caros y comprometer la fidelidad cuando se reducen de escala. GECO integra dos simuladores individuales: PSLF, un simulador basado en Java puede gestionar simulaciones dinámicas de sistemas eléctricos con hasta 30.000 buses, mientras que NS2, escrito en C++ y OTel, ofrece flexibilidad para crear nuevos modelos y protocolos de comunicación.

Uno de los principales retos de la co-simulación es garantizar la sincronización entre las distintas técnicas de simulación utilizadas para los sistemas eléctricos y las redes de comunicaciones. GECO emplea un marco global basado en eventos para abordar este problema con eficacia. A diferencia de los métodos tradicionales de sincronización escalonada en el tiempo, propensos a acumular errores, el marco de GECO sincroniza las simulaciones globalmente de una manera discreta basada en eventos. Este enfoque elimina los errores de sincronización y garantiza una coordinación precisa entre las simulaciones del sistema eléctrico y la red de comunicaciones. Los eventos de simulación dinámica del sistema eléctrico se modelan de forma independiente y se integran con eventos discretos de la red de comunicaciones. Un planificador global procesa los eventos secuencialmente basándose en sus marcas de tiempo, lo que permite el procesamiento inmediato de las solicitudes de interacción entre el sistema eléctrico y los componentes de la red de comunicaciones.

Para facilitar el intercambio de datos entre PSLF y NS2, GECO introduce modelos e interfaces periféricos. El modelo "epcmod" de PSLF y el módulo "Tcl\_PSLF" de NS2 permiten un intercambio de datos fluido entre ambos simuladores. Además, se modifican los protocolos de transporte de comunicaciones para transmitir datos del sistema eléctrico a través de la red de comunicaciones, lo que permite realizar simulaciones integradas de las interacciones entre el sistema eléctrico y la red de comunicaciones.

### 3.2.6 HELICS

HELICS (*Hierarchical Engine for Large-scale Infrastructure Co-Simulation*) es un robusto marco de co-simulación basado en los estándares HLA. Facilita la integración de simuladores escritos en diversos lenguajes de programación como Python, C++, y JAVA, así como de distintos programas de simulación como MATLAB. HELICS está diseñado para funcionar en un entorno escalable y distribuido, lo que lo hace adecuado para simulaciones de infraestructuras a gran escala.

La arquitectura de HELICS gira en torno a un escenario, que se define en un archivo de configuración JSON, en el que se especifican explícitamente los enlaces y parámetros necesarios para las instancias implicadas. En el marco de HELICS, los simuladores se representan como ejecutables genéricos capaces de instanciar múltiples Federates, que a su vez ejecutan modelos físicos específicos definidos en el escenario. La comunicación entre los Federados se facilita a través de núcleos integrados en los simuladores, utilizando un enfoque de publicación/suscripción a través de Brokers. Estos Brokers coordinan el intercambio de datos entre las diferentes Federaciones, permitiendo el despliegue de una arquitectura jerárquica. La Infraestructura de Tiempo de Ejecución (RTI), heredada del estándar HLA, gestiona el papel de Orquestador, garantizando la correcta Regulación y Sincronización del Tiempo tanto en paradigmas de Tiempo Continuo (CT) como de Evento Discreto (DE).

HELICS se desarrolló como software central del proyecto Grid Modernization Laboratory Consortium (GMLC), centrado en la simulación integrada de Transmisión, Distribución y Comunicación en sistemas eléctricos. Fue creado por un equipo de investigadores de varias instituciones de Estados Unidos, entre ellas el Lawrence Livermore National Laboratory (LLNL), el National Renewable Energy Laboratory (NREL) y el Pacific Northwest National Laboratory (PNNL).

Las prioridades de diseño de HELICS están profundamente arraigadas en la facilitación de la colaboración, el mantenimiento de la autonomía entre federados y la provisión de un control de grano fino sobre las operaciones de co-simulación. Estas prioridades influyeron en que HELICS se implementara como una biblioteca en lugar de como un tiempo de ejecución, permitiendo a los federados operar como ejecutables independientes y ofreciendo flexibilidad en la coordinación de tiempos sin imponer requisitos adicionales a las herramientas de simulación participantes.

HELICS sigue una arquitectura modular con una interfaz de programación de aplicaciones (API) unificada para optimizar el rendimiento, acelerar el desarrollo y facilitar el mantenimiento modular. Consta de varias capas: la capa de plataforma, la capa central, la capa de aplicación, la capa de simulador y la capa de interfaz de usuario. Cada capa sirve a un propósito específico, que va desde la compatibilidad del sistema y la gestión de la comunicación hasta la interacción de la simulación y la simplificación de la interfaz de usuario.

Además, HELICS es compatible con las normas de simulación conjunta existentes, como HLA y FMI, lo que permite el acceso directo a una amplia gama de modelos y controles. Sus aplicaciones abarcan diversos campos, especialmente en el ámbito de la energía, permitiendo el estudio de dinámicas transitorias, estudios de planificación a largo plazo, flujos de energía en estado estacionario, comportamiento del mercado e interacciones ciberfísicas. Los desarrollos futuros por parte del equipo de HELICS tienen como objetivo mejorar la usabilidad, apoyar una adopción más amplia y mejorar el rendimiento para despliegues a gran escala.

### 3.2.7 INSPIRE

INSPIRE (*Integrated co-Simulation of Power and ICT systems for Real-time Evaluation*) representa un entorno de simulación de vanguardia que facilita la evaluación exhaustiva tanto de los sistemas de energía como de las redes de comunicaciones dentro de un marco de co-simulación unificado. Aprovecha modelos muy detallados para proporcionar una evaluación exhaustiva del rendimiento de ambos dominios. INSPIRE ofrece flexibilidad al permitir la integración de herramientas adicionales de modelado y simulación para evaluar nuevas aplicaciones utilizando las interfaces proporcionadas.

La arquitectura de INSPIRE es del tipo de simulaciones híbridas, esto proporciona un enfoque genérico para co-simulaciones entre sistemas eléctricos y redes de comunicaciones. La arquitectura consta de tres módulos principales: el núcleo de simulación, la capa de red y la capa de gestión. El núcleo de simulación, basado en la norma IEEE 1516-2010 (HLA evolucionada), facilita la sincronización temporal y de objetos entre simuladores (federados) para garantizar un proceso de simulación reproducible y coherente. La sincronización temporal es crítica por lo que INSPIRE emplea algoritmos conservadores de sincronización temporal para mantener la causalidad temporal.

En INSPIRE, el enfoque de integración implica el acoplamiento de simuladores a nivel de subestación utilizando una representación abstracta denominada unidad de procesamiento de datos de subestación (SSDPU). La SSDPU cumple múltiples funciones, entre ellas vincular los nodos del sistema eléctrico y de la red de comunicaciones basándose en la norma IEC 61850, proporcionar interfaces a los simuladores del sistema eléctrico y de la red de comunicaciones, y permitir la conexión de aplicaciones descentralizadas y componentes adicionales. El enfoque de modelado de las TIC implica la generación de topologías de red de TIC impulsadas por las configuraciones del sistema eléctrico. Los componentes de las posiciones de cada subestación están interconectados con los componentes de la red de comunicaciones, como las unidades de fusión, los controladores de posición, los dispositivos de protección y los disyuntores. La conectividad de área amplia se establece a través de routers WAN, y se supone que las conexiones WAN se realizan mediante fibra óptica.

### 3.2.8 MECSYCO

MECSYCO (*Multi-agent Environment for Complex System CO-simulation*) es un robusto software de co-simulación desarrollado por el equipo SIMBIOT. Ofrece una plataforma para modelar y simular sistemas complejos representándolos como un conjunto de modelos heterogéneos que interactúan, formando lo que se conoce como un ‘multimodelo’. Uno de los principales retos de la co-simulación es la integración de subsistemas distintos, que MECSYCO aborda desde una perspectiva ‘multiagente’. En MECSYCO, un multimodelo se considera una sociedad de modelos heterogéneos que interactúan, cada uno de los cuales actúa como un agente. Este enfoque permite considerar la heterogeneidad a varios niveles, incluidos el modelado multidominio, la interacción/acoplamiento, los formalismos, el software de simulación, los lenguajes y el hardware.

MECSYCO emplea un enfoque de modelado metálico basado en la metáfora del multiagente para describir el multimodelo. Esto implica una representación gráfica y restricciones semánticas para garantizar una descripción clara y sin ambigüedades. El sistema se formaliza utilizando las especificaciones operativas DEVS (*Discrete Event System Specification*), lo que permite la transición de la representación gráfica a la co-simulación ejecutable. El software adopta una estrategia de coordinación descentralizada, en la que cada simulador está asociado a un programador local. Esta coordinación garantiza que la simulación respete las restricciones de causalidad. MECSYCO también proporciona descripciones estructurales y dinámicas del multimodelo, detallando los componentes, las conexiones y la programación de la co-simulación.

En cuanto a las opciones de implementación, MECSYCO se integra con varios simuladores y herramientas de modelado, incluidos los que cumplen la norma FMI. También utiliza middleware como el *Data Distribution Service* (DDS) para la simulación distribuida.

### 3.2.9 MOSAIK

Mosaik es un versátil marco de co-simulación basado en Python diseñado para facilitar la integración de varios simuladores en el ámbito de las Smart Grids, aunque su arquitectura también es aplicable a otros campos. Su arquitectura consta de varios componentes clave y APIs que permiten la comunicación y sincronización sin fisuras entre diferentes simuladores.

Mosaik proporciona una API de simuladores de alto nivel, que ofrece clases abstractas con funcionalidades integradas de comunicación, regulación temporal y sincronización. Estas APIs son agnósticas al lenguaje, permitiendo la integración con simuladores escritos en diferentes lenguajes de programación como Python, C++ y JAVA, así como con varios simuladores de software como MATLAB. Asimismo, ofrece una API de bajo nivel, que permite establecer sockets de red simples para intercambiar datos JSON serializados, ampliando así las capacidades de integración con simuladores. Esta API requiere una meta descripción del simulador, detallando sus parámetros y variables intercambiadas.

La arquitectura de Mosaik incluye dos componentes principales que desempeñan la función de orquestador: el *SimManager* y el *Scheduler*. El *SimManager* es responsable de gestionar el intercambio de datos entre simuladores, incluyendo el arranque de simuladores, la recogida de Instancias de Modelo y la gestión de los procesos de intercambio de datos. Por otro lado, el *Scheduler* regula el tiempo y sincroniza el proceso de simulación, acomodando tanto paradigmas de Tiempo Continuo (TC) como de Evento Discreto (ED) en la versión 3.0 de Mosaik.

La flexibilidad y modularidad de Mosaik lo hacen adecuado para integrar simuladores heterogéneos y aplicaciones de control, permitiendo así la creación de escenarios a gran escala. Emplea el lenguaje MoSL (*Mosaik Specification Language*) para generar escenarios a partir de la información proporcionada por los componentes de simulación. La coordinación de la ejecución de los simuladores se logra mediante un enfoque basado en eventos discretos, lo que permite distintos tamaños de paso y control en tiempo de ejecución.

Además, el núcleo de Mosaik incluye un módulo programador que coordina el intercambio de datos entre simuladores en tiempo de ejecución y rastrea las dependencias entre ellos. Su "componente-API" facilita la integración de las herramientas de simulación existentes, independientemente de su lenguaje de implementación. Además, Mosaik es compatible con varias herramientas de simulación de su ecosistema, como PyPower, PandaPower, PowerFactory y Opal-RT eMEGAsim, a través de interfaces preexistentes o adaptadores FMI conformes con el estándar FMI.

### 3.2.10 VPNET

El entorno de co-simulación VPNET es un sofisticado marco de co-simulación diseñado para integrar *Virtual Test Bed* (VTB) y OPNET, dos poderosas herramientas de simulación para sistemas de energía y redes de comunicación, respectivamente. Al combinar estas herramientas, VPNET ofrece capacidades únicas al permitir el análisis del impacto del rendimiento de la red en el comportamiento, estabilidad y seguridad del sistema eléctrico controlado, así como el estudio de cómo las demandas del sistema eléctrico influyen en el diseño de la red de comunicación.

La estructura general de la co-simulación VPNET se compone de tres elementos principales: VTB, OPNET y el coordinador de co-simulación. Este último, implementado en C#, es crucial para la gestión de pasos de tiempo y el intercambio de datos entre VTB y OPNET. La coordinación se basa en un método de pasos de tiempo, donde el coordinador establece un tiempo de referencia global al que se sincronizan los tiempos locales de simulación de cada simulador. Esto permite una comunicación fluida entre los dos entornos de simulación, a pesar de sus diferencias en la naturaleza de la simulación.

En términos de intercambio de datos, VPNET utiliza interfaces estándar como COM y sockets de *Windows* para permitir la comunicación entre los simuladores y el coordinador. Esto facilita la lectura de variables simuladas, la modificación de parámetros del modelo y la sincronización de tiempos entre VTB y OPNET. Además, VPNET presenta una arquitectura modular que permite la implementación de esquemas de control basados en agentes en el coordinador de co-simulación. Esta característica permite una interacción autónoma entre los agentes asociados a componentes eléctricos en VTB y los nodos de comunicación en OPNET.

Dentro de sus ventajas, esta que permite un análisis integral considerando aspectos tanto del sistema de comunicación como de energía, superando las capacidades de los simuladores individuales. Por parte de los desafíos que aun presenta, esta que requiere un importante esfuerzo de codificación, especialmente debido a las limitaciones actuales para incorporar software de agente directamente. Además, el tiempo total de cálculo puede ser relativamente mayor.

La diversidad de los entornos de co-simulación presentados abarca una amplia gama de aplicaciones y enfoques, desde la simulación de sistemas energéticos distribuidos hasta la integración de sistemas de control y comunicaciones. Cada entorno tiene sus propias herramientas principales y características destacadas, en la **Tabla 3-1** se puede apreciar una visión general de los entornos de co-simulación expuestos anteriormente, en esta se expresa si el entorno es de código abierto o no, el tipo de simulación que utiliza, algunas ventajas y limitaciones que presenta el entorno hasta la fecha, así como las referencias informativas de cada uno. Dentro de estas referencias se puede encontrar desde la documentación propia del entorno de co-simulación hasta algunos proyectos o investigaciones donde estos se han utilizados.

**Tabla 3-1.** Resumen de los principales entornos de co-simulación [10, 12].

Nombre del Entorno	Código Abierto	Tipo de Simulación	Ventajas	Desventajas	Ref. Apl.
<b>CyDER</b>	Si	Eventos discretos	Simulaciones holísticas, análisis fotovoltaico, controles inteligentes, almacenamiento en baterías, simulaciones de carga de vehículos eléctricos.	Desarrollo técnico para FMI, recopilación de datos, QSTS.	[51, 54, 55]
<b>Daccosim-NG</b>	Si	Eventos discretos	Simplifica las co-simulaciones, permite la cooperación entre simuladores (FMUs), proporciona un marco gráfico para diseñar gráficos de cálculo.	Java y JavaFMI, dificultades por distinta naturaleza de simuladores.	[6, 56, 57]
<b>EPOCHS</b>	No	Tiempo Continuo	Integra múltiples simuladores de alta calidad, utiliza componentes middleware para facilitar el intercambio de datos, sincronización temporal.	Dificultades por distinta naturaleza de simuladores (tiempo continuo vs. eventos discretos).	[7, 58, 59]
<b>FNCS</b>	Si	Eventos discretos	Co-simulación sin fisuras de redes eléctricas y de comunicaciones, interfaces para otros lenguajes, entrega puntual y coherente de mensajes entre simuladores.	Críticas por escalabilidad y falta de iteración conjunta dentro de un paso de tiempo.	[8, 60, 61]
<b>GECO</b>	No	Eventos discretos	Aborda el reto de emulación de hardware para sistemas eléctricos a gran escala, elimina errores de sincronización y garantiza una coordinación precisa entre las simulaciones del sistema eléctrico y la red de comunicaciones.	Requiere modificaciones en los protocolos de transporte de comunicaciones.	[62, 63, 64]
<b>HELICS</b>	Si	Eventos discretos	Integración de simuladores en diversos lenguajes de programación, diseño para entornos escalables y distribuidos, arquitectura jerárquica, aplicaciones en diversos campos (especialmente sistemas de energía).	Presenta complejidad en la arquitectura y la implementación.	[5, 9, 65]
<b>INSPIRE</b>	No	Eventos discretos	Flexibilidad para integrar herramientas adicionales de modelado y simulación, algoritmos conservadores de sincronización temporal para mantener la causalidad temporal, generación de topologías de red de TIC impulsadas por las configuraciones del sistema eléctrico.	Requiere conocimientos técnicos para la integración de herramientas adicionales.	[66, 67, 68]
<b>MECSYCO</b>	Si	Eventos discretos	Perspectiva "multiagente" para la co-simulación, integración de subsistemas distintos, representación gráfica y restricciones semánticas para una descripción clara y sin ambigüedades, enfoque de coordinación descentralizada.	Requiere conocimientos de DEVS y modelado multiagente.	[69, 70, 71]
<b>MOSAIK</b>	Si	Eventos discretos	Integración de varios simuladores, APIs agnósticas al lenguaje, flexibilidad y modularidad para integrar simuladores heterogéneos y aplicaciones de control, coordinación de la ejecución de simuladores mediante un enfoque basado en eventos discretos.	Requiere conocimientos de Python y MoSL.	[72, 73, 74]
<b>VPNET</b>	No	Tiempo Continuo	Permite un análisis integral considerando aspectos tanto del sistema de comunicación como de energía, superando las capacidades de los simuladores individuales.	Requiere un importante esfuerzo de codificación, tiempo total de cálculo relativamente mayor.	[75, 76, 77]

### 3.3 Simuladores más utilizados en Co-simulaciones

Como se ha expuesto en capítulos anteriores, las co-simulaciones se han convertido en una herramienta invaluable para comprender y analizar el comportamiento complejo de los sistemas eléctricos. Estas co-simulaciones implican la combinación de varios simuladores para modelar diferentes aspectos de los sistemas eléctricos, como la generación, transmisión y distribución de energía, así como las interacciones con las redes de comunicación y control. En la **Tabla 3-2** se muestran los simuladores más utilizados en co-simulaciones dentro del ámbito de los sistemas eléctricos, con un enfoque en las tendencias de combinación de herramientas de simulación. Es importante destacar que la elección de simuladores puede variar según las necesidades específicas del proyecto y las preferencias individuales de los ingenieros. Por lo tanto, es importante considerar cuidadosamente las características y capacidades de cada simulador al diseñar co-simulaciones en el campo de los sistemas eléctricos.

**Tabla 3-2.** Simuladores más utilizados en co-simulaciones de sistemas eléctricos [10, 11, 12].

POWER AND TRANSMISSION	DISTRIBUTION AND DERs	COMMUNICATIONS AND CONTROL	OTHERS (Buildings, Natural Gas, Transportation, Solvers, Optimization, etc.)
PSS®E/PyPSSE	OpenDSS/PyDSS	OPNET	MATLAB/Simulink
PSCAD	GridLAB-D	GNS3	RTDS
PowerFactory	TESP	NS-2/NS-3	SAInt
PowerWorld	OMF	OMNeT++	BEAM
EMTP-RV	CYMEpy	CPDS	Opal-RT
MATPOWER/MOST	CYMDIST	NetSim	OpenModelica
ANDES	Pylon	GridAPPS-D	NGTransient
InterPSS	PandaPower	Testbed	Synergy-Gas
PYPOWER	PowerFlow	VOLTTRON	EnergyPlus
RSCAD			OCTAVE
			Ptolemy II
			Ochre

Inicialmente, la co-simulación en el ámbito de los sistemas eléctricos se empleaba como una herramienta para fusionar aspectos relacionados con la energía y la comunicación en la red eléctrica. Sin embargo, con el paso del tiempo, se han integrado otras perspectivas, como las simulaciones de precios de mercado y los aspectos de vehículo a red, que han adquirido una gran relevancia en los últimos años. Estos simuladores han demostrado ser de gran utilidad para abordar una variedad de objetivos de investigación mediante co-simulaciones. Entre ellos se incluyen modelar fallos en la energía y evaluar la capacidad de recuperación de la red eléctrica, investigar fallos de dispositivos, simular la carga de vehículos eléctricos para la gestión de la demanda/respuesta, evaluar el impacto de la pérdida de paquetes de comunicación en la red eléctrica, así como explorar diferentes tipos de ataques de inyección de datos.

Por otro lado, en el documento de referencia [10] se presenta un análisis exhaustivo que muestra una variedad de simuladores utilizados en combinación con simuladores de comunicaciones y control en el ámbito de los sistemas eléctricos. Este análisis revela que GridLAB-D, PowerFactory, OpenDSS, OMNeT++ y OpenModelica destacan como los simuladores más frecuentemente combinados con otros. Estas preferencias

pueden atribuirse a la facilidad de extensibilidad, versatilidad y capacidades específicas de cada simulador para modelar aspectos particulares de los sistemas eléctricos.

Se observa que GridLAB-D es el simulador de energía más ampliamente utilizado, seguido de PowerFactory y OpenDSS. GridLAB-D se distingue por proporcionar un entorno completo de gestión que también puede utilizarse para la coordinación de simuladores, lo que justifica su elevado nivel de uso. En cuanto a los simuladores de comunicación y control, se encontró que OMNeT++, NS-3/NS-2 y OPNET fueron los más utilizados, aunque se observó una buena variabilidad en la adopción de estos simuladores. Además, herramientas como Matlab/Simulink y OpenModelica son mencionadas como recursos utilizados para modelar e implementar diversos aspectos relacionados con las simulaciones en sistemas eléctricos.



Figura 3-4. Algunos de los simuladores más utilizados. Elaboración propia.

También se identifica una variedad de combinaciones de simuladores para propósitos de co-simulación como una tendencia emergente. Por ejemplo, OPNET se asocia principalmente con PSCAD y Opal-RT, mientras que OMNeT++ se utiliza con frecuencia en combinación con OpenDSS y PowerFactory. Estas combinaciones reflejan la facilidad de integración entre diferentes simuladores desde el punto de vista de la implementación. La elección del simulador depende del tipo de análisis a realizar y de la complejidad del sistema eléctrico estudiado. La combinación de múltiples simuladores permite un análisis más completo y preciso de los sistemas eléctricos, lo que conduce a avances significativos en la investigación y el desarrollo en este campo.

En otro estudio [11] se comparó la escalabilidad de tres marcos de co-simulación: Mosaik, HELICS y AIOMAS, en un escenario energético multimodelo. Este escenario incluyó simuladores de clima, fotovoltaicos, calefacción/refrigeración de edificios y red eléctrica, con la capacidad de aumentar el número de instancias de modelos para comprender su respuesta en escenarios más grandes. Los resultados revelaron que Mosaik tiene limitaciones en el número de entidades conectadas, deteniéndose en 10k instancias, mientras que HELICS demostró capacidad para escalar hasta 1M de instancias. AIOMAS también mostró escalabilidad similar a HELICS, pero este último superó a AIOMAS en términos de tiempo de simulación en configuraciones multiproceso. En conclusión, Mosaik es adecuado para entornos pequeños/medianos, AIOMAS permite escenarios grandes, pero con limitaciones de escalabilidad, mientras que HELICS ofrece el mejor rendimiento de escalabilidad, especialmente en configuraciones multiproceso, siendo flexible y fácil de implementar, estos aspectos serán explorados en algunos casos de estudios en el capítulo siguiente.

En cuanto a las perspectivas futuras, es crucial seguir monitoreando las tendencias en el uso de simuladores en co-simulaciones en el ámbito de los sistemas eléctricos para identificar nuevas herramientas y enfoques emergentes. La co-simulación seguirá siendo esencial para abordar desafíos futuros, como la integración en bancos de pruebas más grandes y el creciente interés en la incorporación de energías renovables y vehículos eléctricos en la red, lo cual podría constituir un dominio de aplicación relevante para las plataformas de co-simulación.



# 4 CASOS DE ESTUDIO MEDIANTE HELICS

---

*“La mayoría de las ideas fundamentales de la ciencia son esencialmente sencillas y, por regla general pueden ser expresadas en un lenguaje comprensible para todos”.*

*- Albert Einstein -*

En este capítulo se presenta un resumen de los resultados de proyectos, casos de estudio y aplicaciones prácticas desarrollados mediante la herramienta de co-simulación HELICS previamente descrita. En particular, se examinan diversos escenarios donde HELICS demuestra su versatilidad y eficacia, abordando desafíos clave en el ámbito de la energía eléctrica. Es importante destacar que, si bien HELICS tiene sus raíces en aplicaciones dentro del sector eléctrico, su flexibilidad y capacidad de adaptación lo convierten en una herramienta valiosa para una amplia gama de escenarios más allá de la red de energía. Desde la coordinación de infraestructuras múltiples hasta la planificación de ciudades inteligentes, HELICS ofrece un marco sólido para abordar desafíos complejos en diversas áreas.

Este capítulo no solo proporciona una visión general de las aplicaciones prácticas de HELICS, sino que también sirve como un llamado a la comunidad de investigación para explorar y aprovechar aún más el potencial de esta herramienta en campos diversos, fomentando la innovación y la colaboración interdisciplinaria. En el **Anexo B** se proporciona un ejemplo que ilustra la aplicación de co-simulación mediante HELICS en un sistema básico emisor-receptor. En este escenario específico, dos entidades intercambian información de manera sincronizada y controlada temporalmente. Es importante destacar que cada una de estas entidades se ejecuta como un federado individual dentro del entorno de HELICS.

## 4.1 Caso de Estudio 1. Redes de Transmisión-Distribución Multitemporal Escalable (SMTD)

En [78] se presenta un estudio cuya finalidad fue desarrollar un entorno de co-simulación escalable y multitemporal utilizando HELICS mediante Python, el cual integre los simuladores PSS<sup>®</sup>E y GridLAB-D de forma eficaz. Se presentan dos aplicaciones simples para capturar las interacciones Transmisión-Distribución (T&D) en el comportamiento a largo y a corto plazo de la red. Se muestra el impacto del desequilibrio de carga en el sistema de distribución en el margen de estabilidad de voltaje para el comportamiento a largo plazo, y se muestra la respuesta de dos tipos de motores de inducción en el sistema de distribución durante y después de una falla en el sistema de transmisión para el comportamiento a corto plazo. Estos ejemplos validan que los algoritmos de co-simulación implementados funcionan según lo esperado sin problemas numéricos. Además, se discute la generación de datos de medición sintéticos realistas, que es muy útil para el desarrollo y validación de diversas aplicaciones de monitoreo basado en datos y controles basados en datos.

El proceso de co-simulación descrito utiliza un entorno escalable y compatible con la computación en paralelo para analizar la interacción entre los sistemas de transmisión y distribución en la red eléctrica. Se emplean los simuladores PSS<sup>®</sup>E y GridLAB-D, integrados mediante HELICS quien actuó como una interfaz

central para coordinar la comunicación y el intercambio de variables. Esta comunicación facilita el intercambio de datos entre los sistemas, como potencias en buses de carga. Asimismo, permitió sincronizar los pasos de tiempo y mantener la coherencia en la simulación. Además, la configuración de los pasos de tiempo a través de HELICS aseguró una sincronización adecuada en la co-simulación, lo que resultó fundamental para la precisión de los resultados.

Además, se desarrolla una aplicación para generar datos de medición sintéticos, esenciales para validar y mejorar la precisión del modelo. Una vez establecida la co-simulación, se procede a la validación de los resultados y al análisis detallado de las interacciones entre los sistemas de T&D en el contexto de la red eléctrica en evolución.

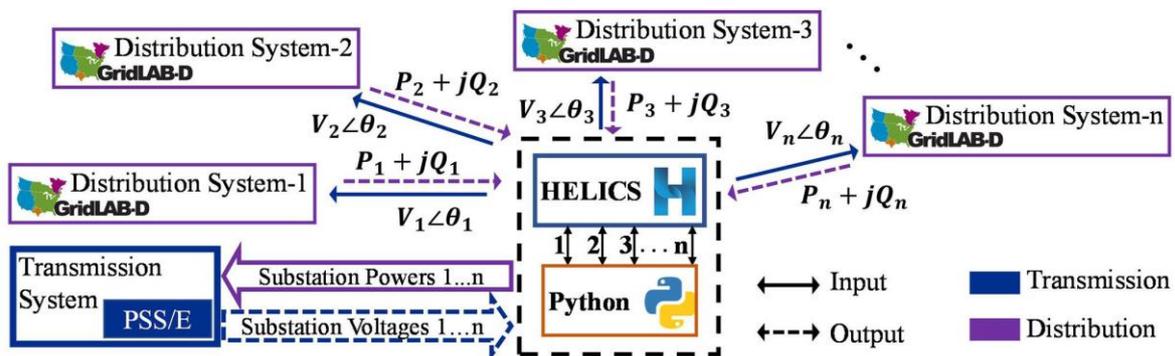


Figura 4-1. Esquema del entorno de co-simulación mediante HELICS para SMTD [78].

En la **Figura 4-1** aparece el diagrama esquemático de un entorno de co-simulación HELICS mediante Python utilizado para análisis de sistemas de energía del estudio presentado. Se pueden identificar diferentes componentes relacionados, entre estos están GridLAB-D como simulador para la distribución eléctrica y PSS®E como simulador para el sistema de transmisión. Los distintos flujos y conexiones representan cómo los datos y las señales de control se mueven entre las partes del sistema para permitir la co-simulación integrada y cómo los sistemas interactúan entre sí y con los sistemas de transmisión.

## 4.2 Caso de Estudio 2. Co-simulación para Análisis de Redes Eléctricas y de Gas Natural

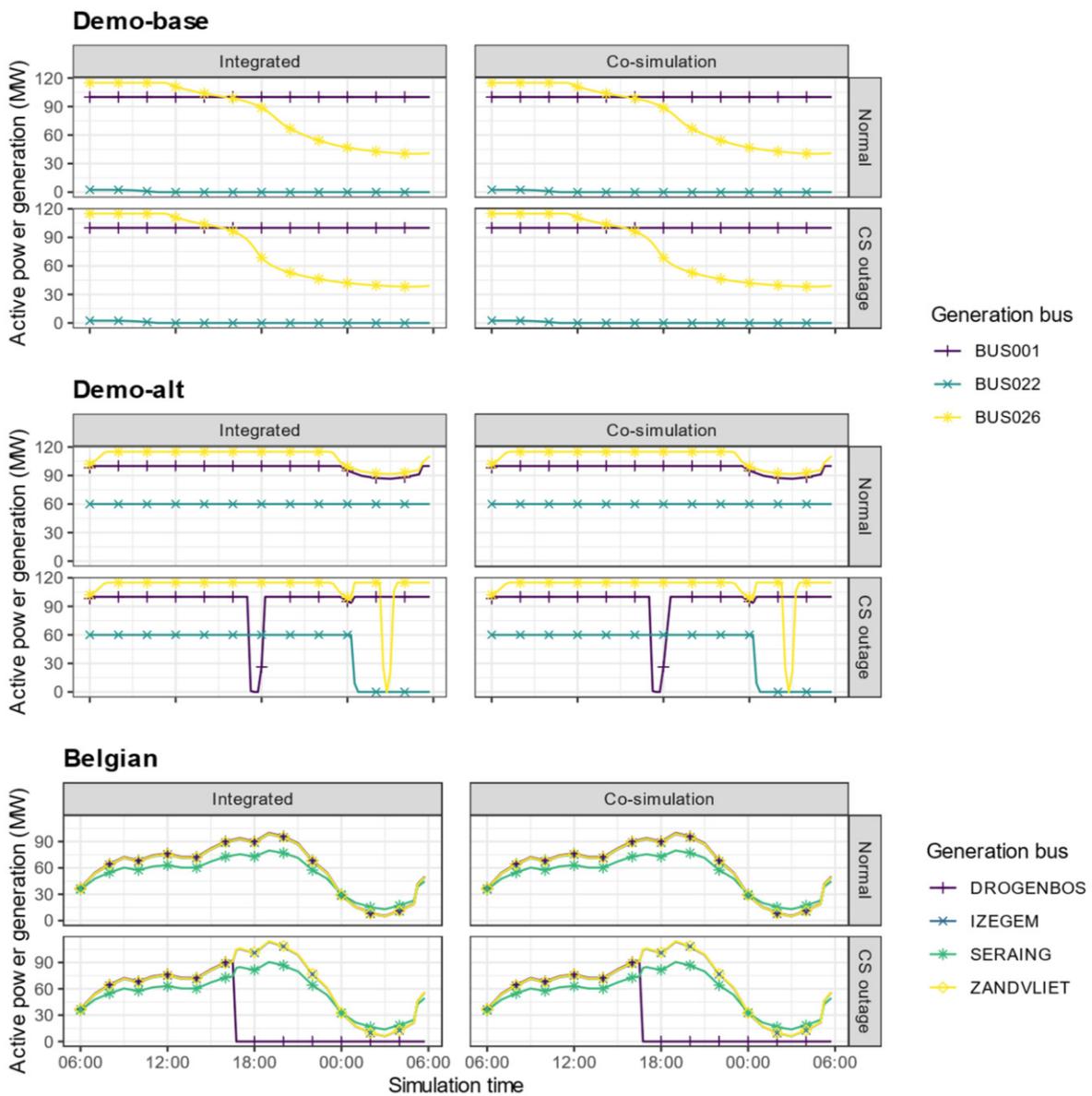
En 2022 dos investigadores del *National Renewable Energy Laboratory* (NREL) y de *Energy Coordination* (encoord) desarrollaron un marco de co-simulación para el modelado acoplado de redes eléctricas y de gas, utilizando la plataforma HELICS y la herramienta SAInt para simular operaciones de sistemas de gas natural y sistemas eléctricos de potencia [79]. Otro avance clave de esta investigación es que se valida y se comparan los resultados del marco de co-simulación con un modelo integrado, lo que brinda confianza adicional en la aproximación de co-simulación para futuras aplicaciones de investigación.

Se observa que al acoplar directamente estos sectores en un modelo unificado puede cerrar la brecha entre los modelos, pero puede ser intensivo en tiempo y recursos computacionales, lo que lo hace desafiante con las herramientas de simulación existentes. Esto se soluciona al aplicar el marco de co-simulación, el cual ofrece un enfoque alternativo que captura las interacciones del sistema sin la carga computacional completa de los modelos integrados.

El proceso de co-simulación empleado se fundamenta en la interacción iterativa y la comunicación entre dos componentes federados: el de energía eléctrica y el de gas. El federado de energía eléctrica estima la potencia

térmica requerida por cada generador en base a las cargas y características del sistema eléctrico, transmitiendo esta información a través de HELICS al federado de gas para su procesamiento. A su vez, este último recibe los datos y determina la cantidad de gas necesaria en cada nodo de la red de gas, considerando las restricciones del sistema. Una vez simulada la entrega de gas, se devuelve al federado eléctrico la potencia térmica disponible para cada generador, actualizada según las restricciones de la red de gas, para ajustar la operación de los generadores. Este intercambio de datos y ajustes se repite en cada paso de tiempo hasta alcanzar la convergencia, logrando un equilibrio donde se cumplen todas las restricciones y se optimiza la operación de ambos sistemas. Este proceso permite la optimización y coordinación entre los sistemas, asegurando una operación eficiente y segura en todo momento.

Se realizan pruebas en seis casos diferentes, incluyendo tres topologías de red distintas y escenarios con interrupciones en la red de gas debido a fallas en compresores. La **Figura 4-2** proporciona la generación de energía activa (en MW) en cada intervalo de la simulación. Por otro lado, la **Figura 4-3** proporciona una visión general similar, centrándose en el flujo de gas en los nodos de salida, que pueden estar acoplados a generadores de gas.



**Figura 4-2.** Generación de energía activa de los generadores en cada red por escenario [79].

Las figuras ilustran un comportamiento casi idéntico en las operaciones de la red de gas en los casos de simulación integrada y co-simulación. El marco de co-simulación captura la dinámica en la presión y el flujo de gas para cada uno de los tres sistemas probados. Además, los cambios en la presión y las entregas nodales debido a la interrupción del compresor en cada una de las redes tienen impactos consistentes en los nodos de cada uno de los sistemas.

Se observa que el enfoque de co-simulación replica el patrón de despacho en los nodos acoplados en el sistema; los resultados en los generadores no acoplados también coinciden en ambos enfoques. Los resultados muestran que la co-simulación puede replicar la capacidad del modelo SAInt para resolver despachos en el sistema, incluso cuando se introducen perturbaciones en el sistema. Además, se comparan los resultados de la co-simulación con el modelo SAInt en tres redes de prueba distintas, incluyendo un modelo belga más grande y realista. Con esto se logra aumentar la confianza en el enfoque de co-simulación, esto debido a que replica con éxito los resultados del modelo integrado en varios escenarios.

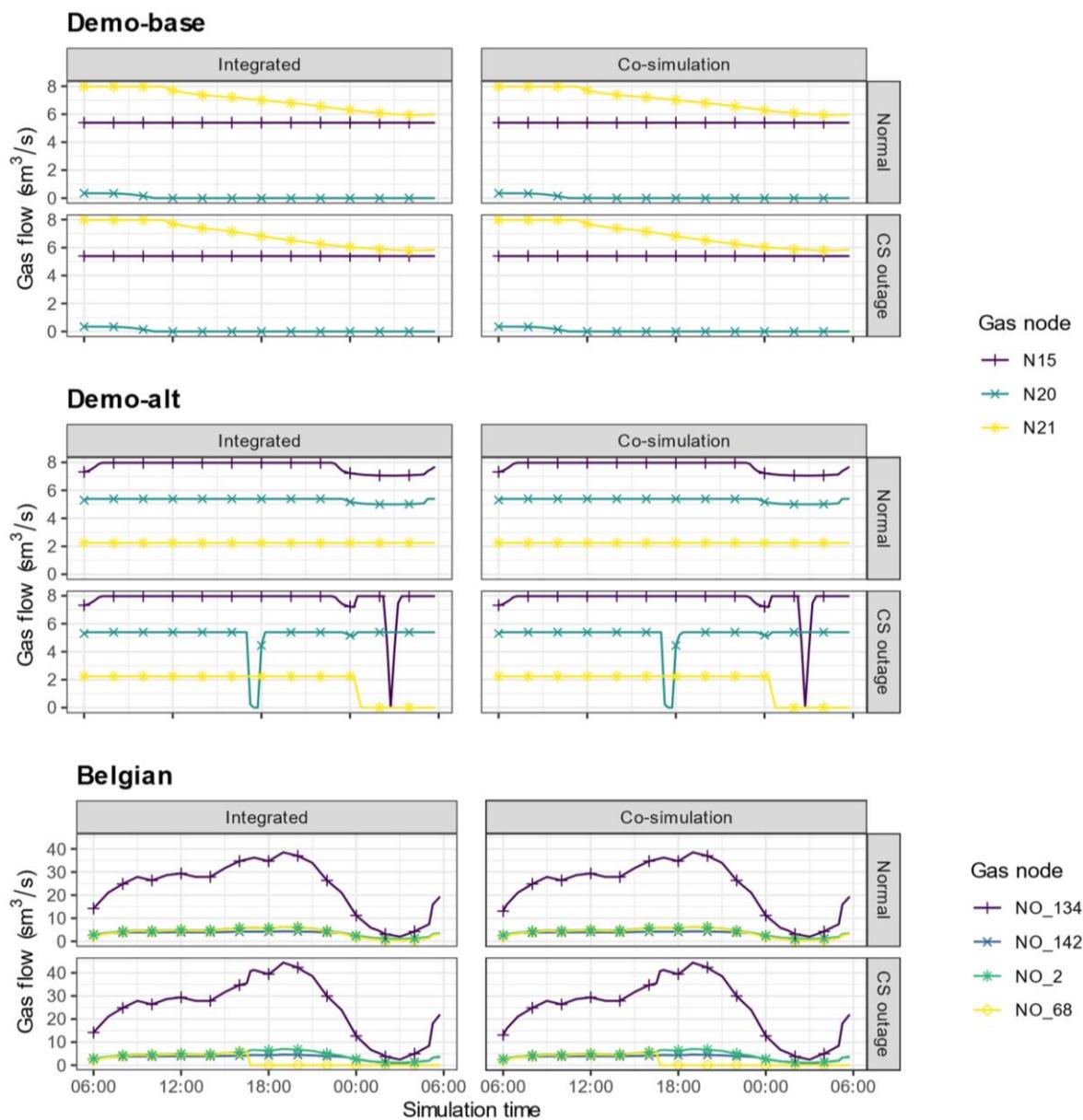
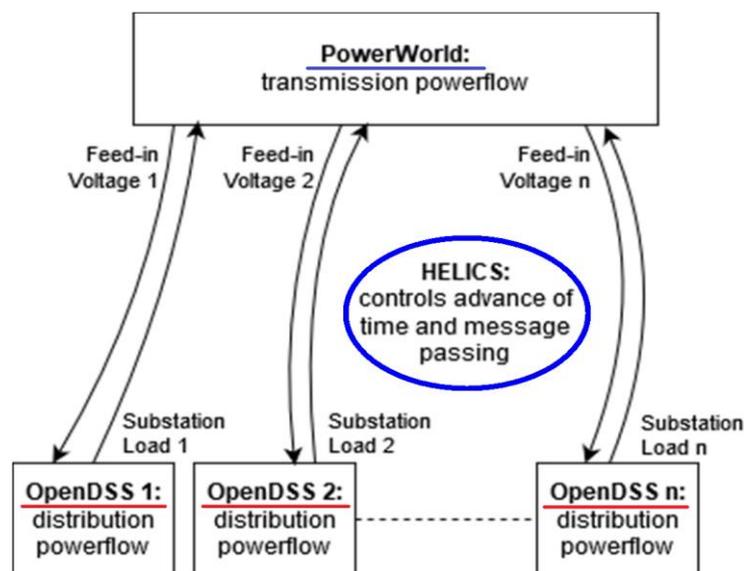


Figura 4-3. Flujo de gas en los nodos de gas para cada red y escenario [79].

### 4.3 Caso de Estudio 3. Co-simulación para el Análisis del Control de Voltaje en Redes Eléctricas

La investigación descrita en este artículo [80] se centra en la co-simulación sintética y realista de sistemas de transmisión y distribución para el control de voltaje en el área metropolitana de Austin, Texas para un día completo de 2012, con intervalos de tiempo de 15 minutos. Se destaca la importancia de probar paradigmas de control y mercado con modelos que integren sistemas de transmisión y distribución, especialmente a medida que los recursos energéticos distribuidos (DER's) desempeñan un papel cada vez más importante en la generación y los servicios de red. Para lograr este objetivo, se utiliza la plataforma de co-simulación HELICS, que permite la comunicación entre los sistemas de distribución y transmisión simulados con diferentes softwares (OpenDSS para la red de distribución y PowerWorld para la red de transmisión) en diferentes núcleos, máquinas y sistemas operativos. En la **Figura 4-4** se puede apreciar el esquemático utilizado en esta investigación.

Cada instancia de OpenDSS y PowerWorld se ejecuta como un federado de HELICS, conectándose a un bus de mensajes para intercambiar publicaciones y suscripciones. Los federados de distribución pasan la carga neta del circuito en el punto de acoplamiento común (PCC) al sistema de transmisión, y viceversa. Se emplean cinco nodos de computación de alto rendimiento, con 36 núcleos físicos o 72 núcleos virtuales, para ejecutar el sistema de distribución. Cada instancia de OpenDSS se ejecuta en su propio núcleo físico, lo que permite un espacio suficiente en cada nodo para manejar procesos en segundo plano. Un núcleo se encarga del broker de HELICS, que contiene el bus de mensajes y asegura que los valores se pasen a los federados correctos. También coordina el avance en el tiempo de las simulaciones, que tienen diferentes tiempos de ejecución. Una máquina Windows separada aloja la simulación del sistema de transmisión de PowerWorld y está vinculada con los federados de distribución a través de HELICS. Una vez que se cargan todos los modelos de OpenDSS, cada paso de tiempo tarda menos de dos minutos en completarse. Dada la falta de controles avanzados o recursos distribuidos en esta simulación, la co-simulación estrechamente acoplada converge en 2 iteraciones o menos para cada paso de tiempo. Se exportan voltajes, cargas y el número de iteraciones por paso de tiempo de ambos sistemas de transmisión y distribución para su análisis.



**Figura 4-4.** Esquema del entorno de co-simulación utilizado en la investigación [80].

La **Figura 4-5** muestra en (a) la simulación del sistema de distribución con un voltaje de alimentación nominal, donde se observan variaciones de voltaje en alrededor del 25% de los buses. Por otro lado, se representa en (b) la co-simulación del sistema de transmisión y distribución, donde se experimentan variaciones de voltaje

en momentos similares, pero en menos buses de distribución. Se destaca que el sistema de transmisión ajusta los voltajes de alimentación y vuelve a resolver el flujo de potencia para adaptarse a las cargas variables a lo largo del día.

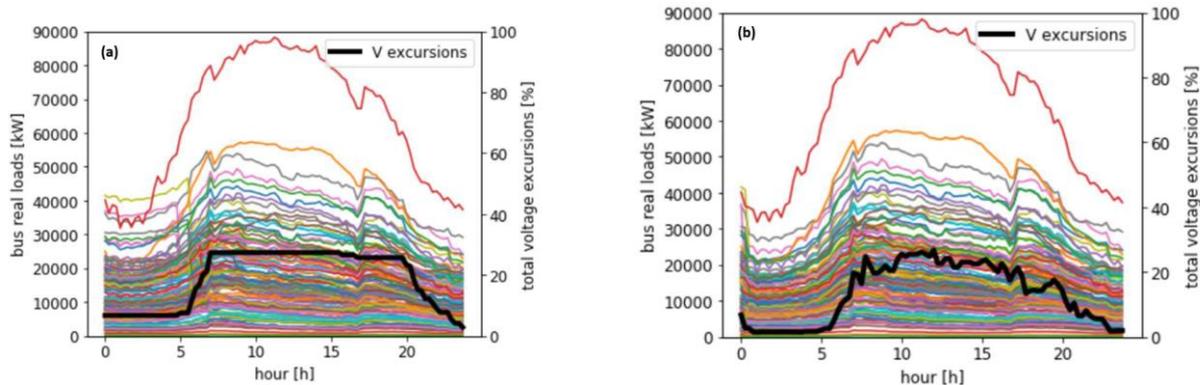


Figura 4-5. Porcentaje de variación cuando se ejecuta en varios casos [80].

La co-simulación de transmisión y distribución resultó en una reducción del 29% en las variaciones de voltaje en comparación con la simulación solo de distribución con voltaje de alimentación nominal. Asimismo, la co-simulación estrechamente acoplada permitió una implementación más precisa de medidas de control de voltaje, ya que los modelos alcanzaron una respuesta convergente a la señal de control en cada paso de tiempo. Además, se destaca la implementación de sistemas de distribución y transmisión mucho más grandes y realistas, así como la captura de diversidad geográficamente relevante en el sistema de distribución. Estos elementos son fundamentales para evaluar métodos de regulación de voltaje y para el benchmarking de sistemas de control en entornos de co-simulación de transmisión y distribución estrechamente acoplados.

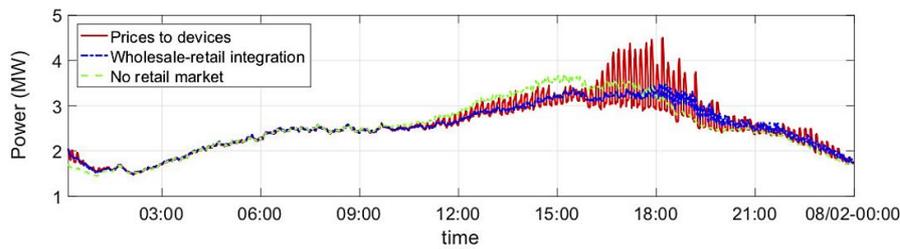
#### 4.4 Caso de Estudio 4. Demanda Sensible al Precio en Interacciones TSO-DSO

La interacción entre los Operadores de Red de Transporte (TSO) y los Operadores de Red de Distribución (DSO) es fundamental en la gestión eficiente de la energía eléctrica. El estudio comparativo que se presenta en [9] examinó el impacto de la demanda sensible al precio en los mercados mayoristas de energía eléctrica, destacando la importancia de la interacción entre los Operadores de Red de Transporte (TSO) y los Operadores de Red de Distribución (DSO). Se encontró que las simulaciones solo considerando la transmisión no lograban capturar oscilaciones clave y diferencias en perfiles de carga en comparación con la demanda fija. Por el contrario, las co-simulaciones que integraban tanto la transmisión como la distribución mostraban una representación más precisa de las interacciones entre los sistemas.

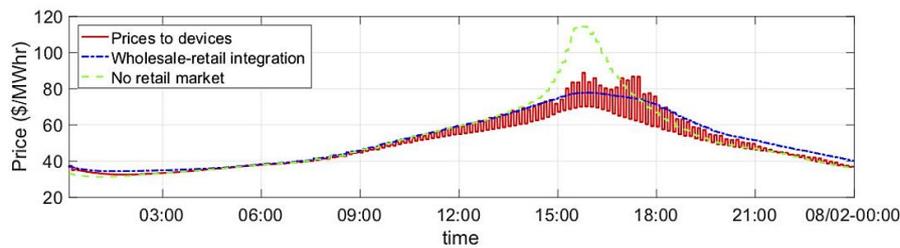
En este también se evaluaron mercados mayoristas (transmisión) y minorista (distribución) integrados con Recursos Energéticos Distribuidos (DER's) sensibles a los precios. Se encontró que la participación de DER's en el mercado energético en tiempo real puede afectar significativamente el sistema de energía física y los precios de liquidación del mercado. Esto subraya la importancia de considerar la respuesta de la demanda a los precios en los mercados energéticos y cómo la co-simulación puede ser una herramienta efectiva para analizar y comprender estas interacciones complejas.

La Figura 4-6 presenta perfiles de potencia real y precios locales marginales relacionados con la integración de DER's en los mercados de energía. Los perfiles de potencia en (a) muestran un comportamiento similar, con oscilaciones de carga correspondientes a las oscilaciones de precio en el caso de precios a dispositivos, las cuales se eliminan con la integración mayorista-minorista. Mientras que en (b) se observa una variación significativa en el precio entre los tres casos de prueba. El caso de referencia sin mercado minorista experimenta un pico de precio en horas de la tarde debido a la congestión en el sistema de transmisión, lo que requiere generación más

costosa. El caso de precios a dispositivos logra reducir el precio, pero con altas oscilaciones. El caso de integración mayorista-minorista no solo reduce el precio pico, sino que mantiene un perfil de precio estable.



(a) Power trajectory



(b) LMP trajectory

**Figura 4-6.** Impacto de DER en mercados mayorista-minorista contra la demanda [9].

Estos gráficos ilustran diferentes aspectos del impacto de los DER's en los sistemas eléctricos y los precios de mercado, mostrando que su participación puede llevar a una mayor eficiencia y potencialmente reducir los costos de energía para los consumidores.

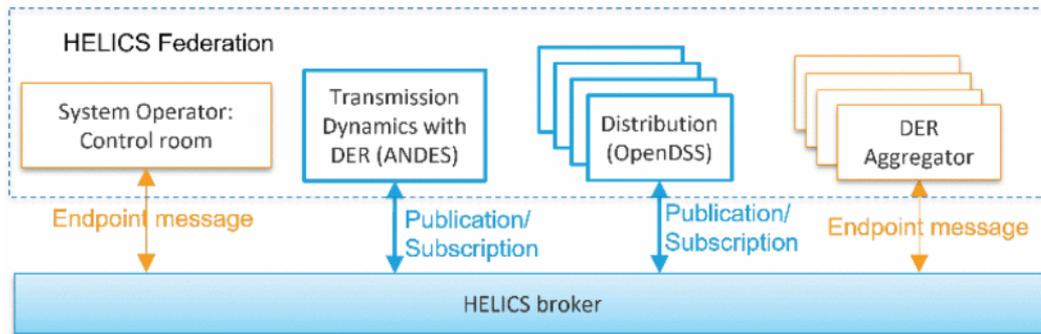
En este estudio, HELICS facilitó las co-simulaciones comparativas entre los mercados mayoristas de energía eléctrica y la demanda sensible al precio. Proporcionó un marco para integrar diferentes herramientas y modelos de simulación, permitiendo el intercambio de datos y la comunicación entre los sistemas de transmisión y distribución. Esto permitió capturar con precisión la dinámica y las interacciones entre los diferentes componentes de la red eléctrica, demostrando la importancia de un enfoque exhaustivo de co-simulación habilitado por HELICS.

## 4.5 Caso de Estudio 5. Evaluación de la Respuesta en Frecuencia en Redes de T&D

Esta investigación describe un marco de co-simulación dinámica de transmisión y distribución (T&D) para la respuesta de frecuencia de recursos energéticos distribuidos (DER) [81]. Se destaca que, aunque existen investigaciones previas sobre la co-simulación dinámica T&D, hay una falta de enfoque en la modelización de la regulación de frecuencia de DER, incluyendo la regulación primaria de frecuencia (PFR), la regulación secundaria de frecuencia (SFR) y las restricciones locales de voltaje en sistemas T&D a gran escala. El diagrama esquemático utilizado se muestra en la **Figura 4-7**, este incluye HELICS, ANDES y OpenDSS los cuales son paquetes/software de código abierto; por lo tanto, el marco de co-simulación propuesta se puede utilizar sin limitaciones de licencia comercial.

Para validar la precisión del marco de co-simulación dinámica T&D propuesto, se creó una red integrada a partir de la red de transmisión IEEE 14-bus y el modelo de secuencia positiva del alimentador IEEE 13-bus como red de distribución (ver **Figura 4-8**). El sistema integrado consta de 26 buses en formato ANDES (el bus principal del alimentador de distribución es el Bus 11 en la red de transmisión IEEE 14-bus). Se agregan cuatro unidades de generación solar distribuida (DPV) con una potencia de salida de 0.5 MW en el sistema de distribución (conectadas a los buses 632, 633, 646 y 671 de la red de 13 buses). Se asume que las 4 unidades de

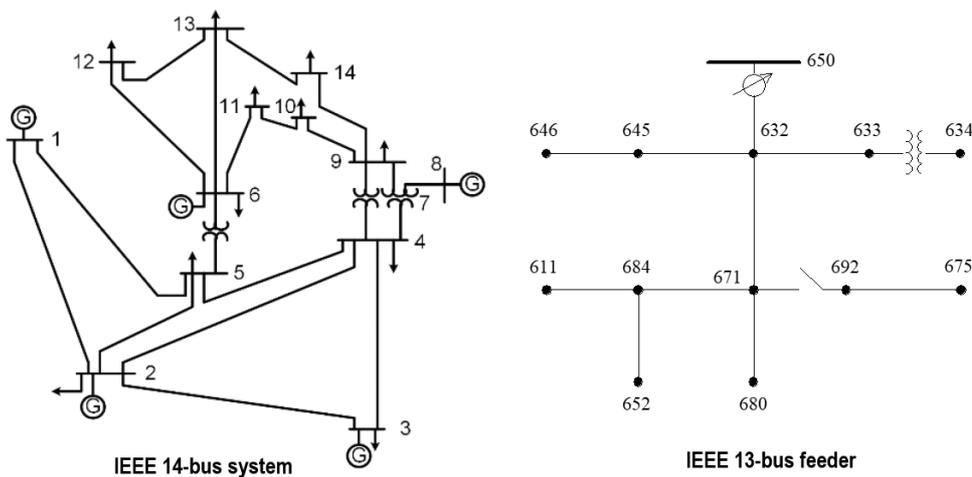
DPV proporcionan el 10% de la respuesta del control automático de generación (AGC) total, con cada DER proporcionando el 2.5% de la respuesta de AGC.



**Figura 4-7.** Esquema del entorno de co-simulación mediante HELICS [81].

Las otras unidades de generación convencionales proporcionan el 90% de la respuesta de AGC, que se distribuye de manera uniforme entre cinco unidades de generación convencionales. Se simula un escenario de desconexión de generación para esta red integrada como referencia para la comparación. Gen 3 con una potencia de salida de 40 MW se desconecta a los 10 segundos. El mismo escenario también se simula utilizando el marco de co-simulación propuesto.

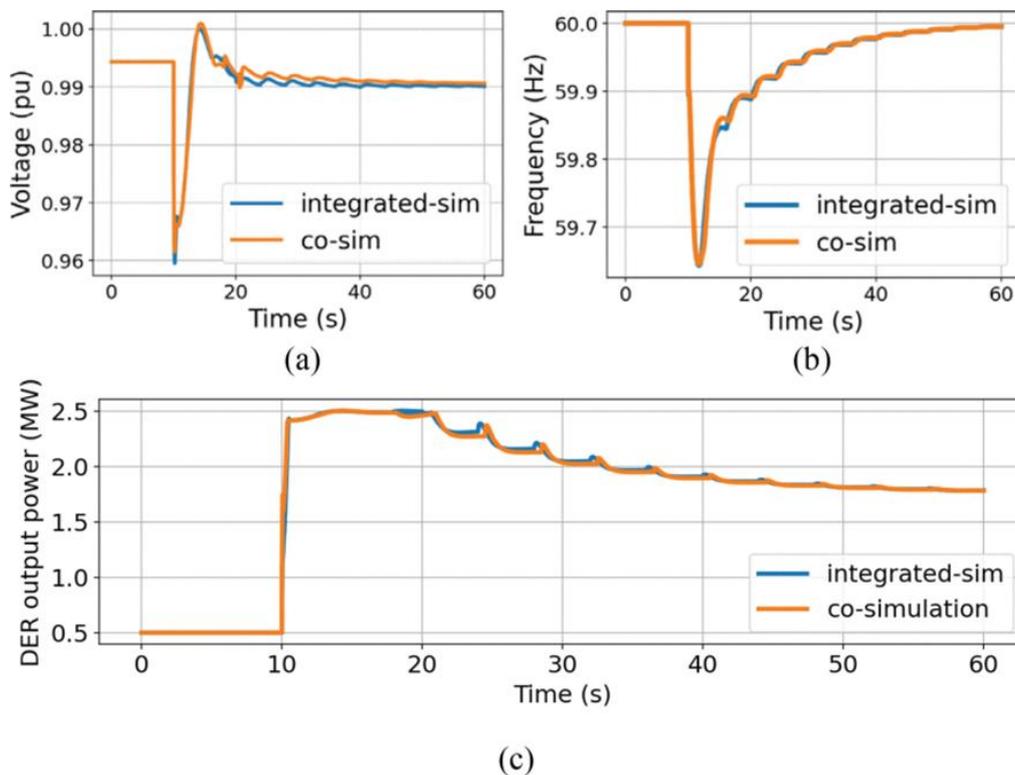
El proceso de co-simulación mediante HELICS para este caso implica la integración de los simuladores ANDES para simulación dinámica de transmisión y OpenDSS para simulación de flujo de potencia en distribución. Los DER se modelan detalladamente en el simulador ANDES con modelos dinámicos de frecuencia, lo que permite capturar su respuesta dinámica en la simulación de frecuencia del sistema de transmisión. La potencia de salida de los DER se intercambia entre ANDES y los simuladores de distribución a través de HELICS, asegurando que su respuesta dinámica se considere en ambos dominios de simulación. HELICS controla implícitamente la sincronización del tiempo de simulación entre los simuladores, garantizando una representación precisa de la interacción entre sistemas de transmisión y distribución con DER.



**Figura 4-8.** Redes de T&D utilizadas en la investigación [81, 82].

La **Figura 4-9** compara los perfiles de (a) voltaje, (b) frecuencia y (c) salida de potencia de DER para la simulación de red integrada de T&D y la co-simulación propuesta. La Figura muestra que las respuestas dinámicas de la frecuencia del sistema principal, el voltaje y la salida de potencia de los DER de la simulación de la red T&D integrada y la co-simulación dinámica T&D propuesta son muy similares. Esto demuestra la

precisión del marco de co-simulación dinámica T&D propuesto. Los perfiles de frecuencia y voltaje del modelo de co-simulación propuesto pueden capturar con precisión la dinámica de la red T&D durante las transitorias del sistema.



**Figura 4-9.** Comparación de diversos perfiles en los escenarios ejecutados [81].

Los resultados del estudio muestran el eficaz desarrollo de un marco de co-simulación dinámica de transmisión y distribución (T&D) para analizar la respuesta de frecuencia de los recursos energéticos distribuidos (DER) en sistemas a gran escala. Se logró una implementación exitosa de la sincronización entre la simulación dinámica de transmisión y la simulación de flujo de potencia de distribución para modelar la regulación de frecuencia de los DER, considerando la intermitencia de su potencia y las restricciones locales de voltaje.

Además, se resalta que la integración de simuladores de sistemas de potencia de transmisión y distribución a través de HELICS facilita una comunicación eficiente y una sincronización precisa entre los diferentes simuladores, lo que permite el estudio de la interacción dinámica entre los DER y las redes de energía. Finalmente, se subraya que el marco de co-simulación propuesto proporciona una herramienta valiosa para investigar y comprender la respuesta de frecuencia de los DER en sistemas T&D, lo que es crucial para la planificación y operación eficiente de las redes eléctricas con altas penetraciones de energías renovables distribuidas.



# 5 CONCLUSIONES

---

*“El optimismo es la fe que conduce al logro. Nada puede hacerse sin esperanza y confianza”.*

*- Helen Keller -*

Después de realizar un análisis exhaustivo del estado actual de la co-simulación en el ámbito de los sistemas de energía eléctrica, así como una revisión de los principales entornos de co-simulación disponibles y estudiar de varios casos de estudio donde se utiliza HELICS de forma destacada, se pueden extraer conclusiones significativas que destacan la importancia y el potencial de esta técnica en la ingeniería eléctrica moderna.

Es fundamental resaltar que las conclusiones expuestas no solo aportan al conjunto de conocimientos actual en el área, sino que también ofrecen pautas que orientan investigaciones futuras. En definitiva, estas conclusiones se convierten en un recurso invaluable para quienes en un futuro participarán en el desarrollo de co-simulaciones para el análisis y estudio de sistemas eléctricos complejos.

En primer lugar, la revisión detallada del estado actual de la co-simulación ha permitido obtener una comprensión completa de los fundamentos teóricos, los componentes, la normativa, las aplicaciones prácticas y las últimas tendencias en esta área. Se puede resaltar su capacidad de combinar modelos de diferentes dominios y disciplinas de ingeniería para proporcionar una visión más completa del comportamiento de sistemas complejos y facilitar la toma de decisiones informada en el diseño, análisis y optimización de sistemas ingenieriles.

Se ha identificado que la co-simulación es una herramienta invaluable para integrar modelos heterogéneos y realizar análisis avanzados en sistemas eléctricos complejos. Asimismo, se ha demostrado que la co-simulación es una técnica efectiva para mejorar la eficiencia energética y la confiabilidad del sistema eléctrico. A través de la integración de modelos de los diversos componentes se pueden evaluar con mayor precisión el comportamiento del sistema y las posibles estrategias de optimización.

La evaluación de diferentes entornos de co-simulación ha resaltado la importancia de seleccionar la plataforma adecuada según las necesidades específicas del proyecto y los requisitos de precisión y rendimiento. Tanto los entornos comerciales como las herramientas de código abierto ofrecen ventajas únicas, pero es crucial considerar aspectos como la interoperabilidad, la flexibilidad y el soporte técnico al elegir una plataforma.

HELICS se destaca como una herramienta líder en el campo de la co-simulación, gracias a su enfoque en la interoperabilidad y la escalabilidad, especialmente para sistemas eléctricos y de comunicaciones. Su arquitectura federada de alto rendimiento permite la simulación de sistemas distribuidos a gran escala, abordando problemas complejos de ingeniería con eficacia y precisión.

Una de las principales ventajas de HELICS es su capacidad para integrar diversos tipos de simuladores y modelos, permitiendo la interoperabilidad entre diferentes herramientas de simulación y facilitando la colaboración entre equipos de investigación y desarrollo. Además, a pesar de que no posee una interfaz amigable

como otras plataformas, HELICS ofrece una amplia gama de funcionalidades y APIs que simplifican el proceso de modelado, simulación y análisis de sistemas energéticos complejos.

Los casos de estudio presentados han proporcionado ejemplos concretos de cómo la co-simulación se ha aplicado en una variedad de contextos, incluyendo el análisis de la estabilidad del sistema de transmisión y distribución, así como la interacción entre una red eléctrica y un sistema de gasoductos, entre otros. Estos casos han demostrado el valor de la co-simulación como una herramienta para abordar desafíos reales y tomar decisiones informadas en el campo de la ingeniería eléctrica.

Se aconseja a los investigadores a mantenerse actualizados sobre las tendencias y avances más recientes en el ámbito de la co-simulación en sistemas eléctricos, con la finalidad de incorporar las innovaciones más actuales en sus proyectos y enfoques estratégicos. Se ha identificado que el desarrollo futuro de la co-simulación se verá influenciado por avances en áreas como la inteligencia artificial y el modelado de sistemas de energía eléctrica basados en datos. Debido a que estas tecnologías tienen el potencial de mejorar la precisión y la eficiencia de la co-simulación, abriendo nuevas oportunidades para la investigación y la innovación en el campo.

# REFERENCIAS

---

- [1] H. Gould, J. Tobochnik y C. Wolfgang, *An Introduction to Computer Simulation Methods: Applications To Physical Systems*, Carolina del Sur: CreateSpace, 2017, p. 12.
- [2] A. M. Law, *Simulation Modeling and Analysis*, vol. 5, New York: McGraw-Hill Education, 2015.
- [3] C. Gomes, C. Thule, D. Broman, P. G. Larsen y H. Vangheluwe, *Co-Simulation: A Survey*, vol. 51, A. C. Surv., Ed., New York, NY, USA, NY: Association for Computing Machinery, 2018.
- [4] Electric Power Research Institute - EPRI, «An Overview of Co-Simulation Platforms for Transmission Planning,» *Technology Innovation*, 11 Dec 2019.
- [5] T. D. Hardy, B. Palmintier, P. L. Top, D. Krishnamurthy y J. C. Fuller, «Co-Simulation Overview - HELICS documentation,» 2021. [En línea]. Available: [https://github.com/GMLC-TDC/HELICS/blob/main/docs/user-guide/co-simulation\\_overview.md](https://github.com/GMLC-TDC/HELICS/blob/main/docs/user-guide/co-simulation_overview.md).
- [6] J. Évora Gómez, J. J. Hernández Cabrera, J.-P. Tavella, S. Vialle, E. Kremers y L. Frayssinet, «Daccosim NG: co-simulation made simpler and faster,» *13th International Modelica Conference 2019*, Mar 2019.
- [7] K. Hopkinson, X. Wang, R. Giovanini, J. Thorp, K. Birman y D. Coury, «EPOCHS: a platform for agent-based electric power and communication simulation built from commercial off-the-shelf components,» *IEEE Transactions on Power Systems*, vol. 21, n° 2, pp. 548-558, 2006.
- [8] S. Ciraci, J. Daily, J. Fuller, A. Fisher, L. Marinovici y K. Agarwal, «FNCS: A framework for power system and communication networks co-simulation,» de *Proceedings of the symposium on theory of modeling & simulation-DEVS integrative*, San Diego, Society for Computer Simulation International, 2014.
- [9] T. D. Hardy, B. Palmintier, P. L. Top, D. Krishnamurthy y J. C. Fuller, «HELICS: A Co-Simulation Framework for Scalable Multi-Domain Modeling and Analysis,» *IEEE Access*, vol. 12, pp. 24325-24347, Feb 2024.
- [10] P. Mihal, M. Schvarcbacher, B. Rossi y T. Pitner, «Smart grids co-simulations: Survey & research directions,» *Sustainable Computing: Informatics and Systems*, vol. 35, p. 100726, 2022.
- [11] L. Barbierato, P. Rando Mazzarino, M. Montarolo, A. Macii, E. Patti y L. Bottaccioli, «A comparison study of co-simulation frameworks for multi-energy systems: the scalability problem,» *Energy Informatics*, vol. 5, n° 4, pp. 53-79, 12 Dic 2022.
- [12] M. Vogt, F. Marten y M. Braun, «A survey and statistical analysis of smart grid co-simulations,» *Applied energy*, vol. 222, pp. 67-78, 15 Jul 2018.
- [13] P. Palensky, A. A. Van Der Meer, C. D. Lopez, A. Joseph y K. Pan, «Cosimulation of Intelligent Power Systems: Fundamentals, Software Architecture, Numerics, and Coupling,» *IEEE Industrial Electronics*

- Magazine*, vol. 11, n° 1, pp. 34-50, 21 Mar 2017.
- [14] L. G. Birta y G. Arbez, *Modelling and Simulation: Exploring Dynamic System Behaviour*, 3° ed., Springer, Ed., Switzerland, 2019.
- [15] R. Fujimoto, «Parallel and distributed simulation systems,» de *Proceeding of the 2001 Winter Simulation Conference (Cat. No.01CH37304)*, vol. 1, Arlington, VA: IEEE, 2001, pp. 147-157.
- [16] A. Godfraind, «Co-simulation platform for cyber-physical transmission systems,» Belgium, 2021.
- [17] A. Muzy, L. Touraille, H. Vangheluwe, O. Michel, M. K. Traoré y D. R. C. Hill, «Activity regions for the specification of discrete event systems,» *Proceedings of the 2010 Spring Simulation Multiconference*, vol. 136, p. 7, 2010.
- [18] B. Van Acker, J. Denil, H. Vangheluwe y P. De Meulenaere, «Generation of an Optimised Master Algorithm for FMI Co-simulation,» *Simulation Series*, vol. 47, p. 946–953, 2015.
- [19] S. V. Mierlo, H. Vangheluwe, S. Breslav, R. Goldstein y A. Khan, «Extending Explicitly Modelled Simulation Debugging Environments with Dynamic Structure,» *ACM Trans. Model. Comput. Simul.*, vol. 30, n° 1, p. 25, 2020.
- [20] B.-S. Lee, W. Cai, S. Turner y L. Chen, «Adaptive dead reckoning algorithms for Distributed Interactive Simulation,» *International Journal of Simulation: Systems, Science & Technology*, vol. 1, n° 1-2, pp. 21-34, 2000.
- [21] I. Hafner y N. Popper, «An Overview of the State of the Art in Co-Simulation and Related Methods,» *Simulation Notes Europe*, vol. 31, n° 4, pp. 185-200, Dec 2021.
- [22] V. H. Nguyen, T. L. Nguyen, Y. Besanger y T. Tran-Quoc, «On Conceptual Structuration and Coupling Methods of Co-Simulation Frameworks in Cyber-Physical Energy System Validation,» *Energies*, vol. 10, pp. 1977-1996, Nov 2017.
- [23] J. W. Forrester, «Industrial dynamics,» *Journal of the Operational Research Society*, vol. 48, n° 10, pp. 1037-1041, 1997.
- [24] J. Banks, J. S. Carson, B. L. Nelson y D. Nicol, *Discrete-Event System Simulation*, 5 ed., New Jersey: Prentice Hall, 2010.
- [25] F. Schloegl, L. Fischer, S. Lehnhoff, R. Rosen y J. Wehrstedt, «Co-simulation techniques in assistance systems for process control,» *2017 IEEE 15th International Conference on Industrial Informatics (INDIN)*, pp. 971-976, Jul 2017.
- [26] S. Scherfke, «aiomas Documentation - Release 1.0.2,» May 2016. [En línea]. Available: <https://readthedocs.org/projects/aiomas/downloads/pdf/1.0.2/>.
- [27] R. Yang y J. Xu, «Computing at Massive Scale: Scalability and Dependability Challenges,» de *2016 IEEE Symposium on Service-Oriented System Engineering (SOSE)*, Oxford, 2016, pp. 386-397.
- [28] K. Balasubramaniam, S. Plathottam, S.-I. Yim, N. Kang, K. Jhala, R. Bhattarai y S. Zhao, «Co-Simulation of Transmission and Distribution Systems—From Modeling to Software Development,» *IEEE Access*, vol. 10, pp. 127061-127072, 2022.

- [29] Open Simulation Platform, «Co-simulation,» Technische Universität Hamburg-Harburg, 2020. [En línea]. Available: <https://opensimulationplatform.com/co-simulation/>.
- [30] M. H. Cintuglu, O. A. Mohammed, K. Akkaya y A. S. Uluagac, «A Survey on Smart Grid Cyber-Physical System Testbeds,» *IEEE Communications Surveys & Tutorials*, vol. 19, n° 1, pp. 446-464, 2017.
- [31] P. Pourghasem Gavvani, S. Baghbannovin, S. M. Mohseni-Bonab y I. Kamwa, «Distributed Energy Resources Management System (DERMS) and Its Coordination with Transmission System: A Review and Co-Simulation,» *Energies*, vol. 17, n° 6, p. 1353, Mar 2024.
- [32] X. Li, Q. Huang y D. Wu, «Distributed large-scale co-simulation for IoT-aided smart grid control,» *IEEE Access*, vol. 5, pp. 19951-19960, 2017.
- [33] H. Mirtaheri, G. Chicco, V. del Razo y H.-A. Jacobsen, «A framework for control and co-simulation in distribution networks applied to electric vehicle charging with Vehicle-Originating-Signals,» *2016 IEEE International Energy Conference (ENERGYCON)*, pp. 1-6, Apr 2016.
- [34] P. Palensky, E. Widl, M. Stifter y A. Elsheikh, «Modeling Intelligent Energy Systems: Co-Simulation Platform for Validating Flexible-Demand EV Charging Management,» *IEEE Transactions on Smart Grid*, vol. 4, n° 4, pp. 1939-1947, Dec 2013.
- [35] C. Shum, W. Lau, K. L. Lam, Y. He, H. Chung, N. C. Tse, K. Tsang y L. Lai, «The development of a smart grid co-simulation platform and case study on Vehicle-to-Grid voltage support application,» *2013 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 594-599, Oct 2013.
- [36] Q. Wang, W. Tai, Y. Tang, Y. Liang, L. Huang y D. Wang, «Architecture and Application of Real-time Co-simulation Platform for Cyber-physical Power System,» *2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, pp. 81-85, Aug 2017.
- [37] M. Garau, G. Celli, E. Ghiani, G. G. Soma, F. Pilo y S. Corti, «ICT reliability modelling in co-simulation of smart distribution networks,» *2015 IEEE 1st International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI)*, pp. 365-370, Sep 2015.
- [38] M. Otte, F. Leimgruber, R. Bründlinger, S. Rohjans, A. Latif y T. I. Strasser, «Hardware-in-the-Loop Co-Simulation Based Validation of Power System Control Applications,» *2018 IEEE 27th International Symposium on Industrial Electronics (ISIE)*, pp. 1229-1234, 2018.
- [39] B. M. Kelley, P. Top, S. G. Smith, C. S. Woodward y L. Min, «A federated simulation toolkit for electric power grid and communication network co-simulation,» *2015 Workshop on Modeling and Simulation of Cyber-Physical Energy Systems (MSCPES)*, pp. 1-6, 2015.
- [40] N. Duan, N. Yee, B. Salazar, J.-Y. Joo, E. Stewart y E. Cortez, «Cybersecurity Analysis of Distribution Grid Operation with Distributed Energy Resources via Co-Simulation,» *2020 IEEE Power & Energy Society General Meeting (PESGM)*, pp. 1-5, 2020.
- [41] T. Hess, J. Dickert y P. Schegner, «Multivariate power flow analyses for smart grid applications utilizing Mosaik,» *2016 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, pp. 1-6, 2016.
- [42] S. Lehnhoff, O. Nannen, S. Rohjans, F. Schlogl, S. Dalhues, L. Robitzky, U. Hager y C. Rehtanz, «Exchangeability of power flow simulators in smart grid co-simulations with mosaik,» *2015 Workshop*

- on Modeling and Simulation of Cyber-Physical Energy Systems (MSCPES)*, pp. 1-6, 2015.
- [43] S. Hanif, L. Marinovici, M. Pelton, T. Hardy y T. E. McDermott, «Simplified Transactive Distribution Grids for Bulk Power System Mechanism Development,» *2021 IEEE Power & Energy Society General Meeting (PESGM)*, pp. 1-5, 2021.
- [44] L. Bottaccioli, A. Estebasari, E. Pons, E. Bompard, E. Macii, E. Patti y A. Acquaviva, «A Flexible Distributed Infrastructure for Real-Time Cosimulations in Smart Grids,» *IEEE Transactions on Industrial Informatics*, vol. 13, nº 6, pp. 3265-3274, 2017.
- [45] D. Bytschkow, M. Zellner y M. Duchon, «Combining SCADA, CIM, GridLab-D and AKKA for smart grid co-simulation,» *2015 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1-5, 2015.
- [46] M. H. Syed, P. Crolla, G. M. Burt y J. K. Kok, «Ancillary service provision by demand side management: A real-time power hardware-in-the-loop co-simulation demonstration,» *2015 International Symposium on Smart Electric Distribution Systems and Technologies (EDST)*, pp. 492-498, 2015.
- [47] S. Mallapuram, W. Yu, P. Moulema, D. Griffith, N. Golmie y F. Liang, «An Integrated Simulation Study on Reliable and Effective Distributed Energy Resources in Smart Grid,» *Proceedings of the International Conference on Research in Adaptive and Convergent Systems*, p. 140–145, 2017.
- [48] J. J. Chromik, C. Pilch, P. Brackmann, C. Duhme, F. a. G. A. Everinghoff, T. Teodorowicz, J. Wieland, B. R. Haverkort y A. Remke, «Context-aware local Intrusion Detection in SCADA systems: A testbed and two showcases,» *2017 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 467-472, Oct 2017.
- [49] M. Abdelrahman, I. Kharchouf, T. Nguyen y O. Mohammed, «A Hybrid Physical Co-Simulation Smart Grid Testbed for Testing and Impact Analysis of Cyber-Attacks on Power Systems: Framework and Attack Scenarios,» *Energies*, vol. 16, nº 23, pp. 7771-7796, 2023.
- [50] W. Wang, M. Cai, X. Fang y C. Irwin, «Impact of Open Communication Networks on Load Frequency Control with Plug-In Electric Vehicles By Cyber-Physical Dynamic Co-simulation,» *2023 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1-5, Jan 2023.
- [51] T. S. Nouidui, J. Coignard, C. Gehbauer, M. Wetter, J.-Y. Joo y E. Vrettos, «CyDER – an FMI-based co-simulation platform for distributed energy resources,» *Journal of Building Performance Simulation*, vol. 12, nº 5, pp. 566-579, 2019.
- [52] J. Évora, «Daccosim NG: from OpenModelica to FMU - YouTube,» Mar 2019. [En línea]. Available: <https://www.youtube.com/watch?v=Cqnx1obNg5E>.
- [53] Pacific Northwest National Lab. (PNNL), «Introducing FNCS: Framework for Network Co-Simulation,» Oct 2014. [En línea]. Available: <https://www.osti.gov/sciencecinema/biblio/1185200>.
- [54] C. Gehbauer, «CyDER: A Cyber Physical Co-simulation Platform for Distributed Energy Resources in Smartgrids,» LBL Publications, 2022.
- [55] X. Lin, «Co-simulation of Cyber-Physical Systems Using DEVS and Functional Mockup Units,» *ASU Electronic Theses and Dissertations*, 2021.
- [56] V. Galtier, S. Vialle, C. Dad, J.-P. Tavella, J.-P. Lam-Yee-Mui y G. Plessis, «FMI-based distributed multi-simulation with DACCOSIM,» *Proceedings of the Symposium on Theory of Modeling & Simulation:*

- DEVS Integrative M&S Symposium*, p. 39–46, 2015.
- [57] C. Dad, S. Vialle, M. Caujolle, J.-P. Tavella y M. Ianotto, «Scaling of Distributed Multi-Simulations on Multi-Core Clusters,» *2016 IEEE 25th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*, pp. 142-147, 2016.
- [58] J. S. Thorp, X. Wang, K. M. Hopkinson, D. Coury y R. Giovanini, «Agent Technology Applied to the Protection of Power Systems,» *Autonomous Systems and Intelligent Agents in Power System Control and Operation*, pp. 115-154, 2003.
- [59] D. Bhor, K. Angappan y K. M. Sivalingam, «Network and power-grid co-simulation framework for Smart Grid wide-area monitoring networks,» *Journal of Network and Computer Applications*, vol. 59, pp. 274-284, 2016.
- [60] J. Daily, S. Ciraci, J. Fuller, L. Marinovici, A. Fisher, C. Lo y M. Hauer, «Framework for Network Co-Simulation,» Richland, WA (United States), 2014.
- [61] A. Razaq, B. Pranggono, H. Tianfield y H. Yue, «Simulating smart grid: Co-simulation of power and communication network,» *2015 50th International Universities Power Engineering Conference (UPEC)*, pp. 1-6, Sep 2019.
- [62] H. Lin, S. S. Veda, S. S. Shukla, L. Mili y J. Thorp, «GECO: Global Event-Driven Co-Simulation Framework for Interconnected Power System and Communication Network,» *IEEE Transactions on Smart Grid*, vol. 3, n° 3, pp. 1444-1456, May 2012.
- [63] H. Lin, S. Sambamoorthy, S. Shukla, J. Thorp y L. Mili, «Power system and communication network co-simulation for smart grid applications,» *ISGT 2011*, pp. 1-6, 2011.
- [64] H. Lin, S. Sambamoorthy, S. Shukla, J. Thorp y L. Mili, «A study of communication and power system infrastructure interdependence on PMU-based wide area monitoring and protection,» *2012 IEEE Power and Energy Society General Meeting*, pp. 1-7, Jul 2012.
- [65] B. Palmintier, D. Krishnamurthy, P. Top, S. Smith, J. Daily y J. Fuller, «Design of the HELICS high-performance transmission-distribution-communication-market co-simulation framework,» *2017 Workshop on Modeling and Simulation of Cyber-Physical Energy Systems (MSCPES)*, pp. 1-6, Apr 2017.
- [66] H. Georg, S. C. Müller, N. Dorsch, C. Rehtanz y C. Wietfeld, «INSPIRE: Integrated co-simulation of power and ICT systems for real-time evaluation,» *2013 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 576-581, Oct 2013.
- [67] S. C. Müller, H. Georg, C. Rehtanz y C. Wietfeld, «Hybrid simulation of power systems and ICT for real-time applications,» *2012 3rd IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe)*, pp. 1-7, Oct 2012.
- [68] H. Georg, S. C. Müller, C. Rehtanz y C. Wietfeld, «Analyzing Cyber-Physical Energy Systems: The INSPIRE Cosimulation of Power and ICT Systems Using HLA,» *IEEE Transactions on Industrial Informatics*, vol. 10, n° 4, pp. 2364-2373, 2014.
- [69] B. Camus, T. Paris, J. Vaubourg, Y. Presse, C. Bourjot, L. Ciarletta y V. Chevrier, «MECSYCO: a Multi-agent DEVS Wrapping Platform for the Co-simulation of Complex Systems,» 2016.
- [70] J. Vaubourg, Y. Presse, B. Camus, C. Bourjot, L. Ciarletta, V. Chevrier, J.-P. Tavella y H. Morais, «Multi-agent Multi-Model Simulation of Smart Grids in the MS4SG Project,» *Advances in Practical Applications*

- of Agents, Multi-Agent Systems, and Sustainability: The PAAMS Collection*, pp. 240-251, May 2015.
- [71] T. Paris, «Complex System Modeling by Composition. A Hierarchical Approach for Heterogeneous Components Co-simulation,» HAL Open Science, 2019.
- [72] S. Schütte, S. Scherfke y M. Tröschel, «Mosaik: A framework for modular simulation of active components in Smart Grids,» *2011 IEEE First International Workshop on Smart Grid Modeling and Simulation (SGMS)*, pp. 55-60, 2011.
- [73] M. Büscher, A. Claassen, M. Kube, S. Lehnhoff, K. Piech, S. Rohjans, S. Scherfke, C. Steinbrink, J. Velasquez, F. Tempez y Y. Bouzid, «Integrated Smart Grid simulations for generic automation architectures with RT-LAB and MOSAIK,» de *2014 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Venice, Italy, 2014, pp. 194-199.
- [74] A. M. Kosek, O. Lünsdorf, S. Scherfke, O. Gehrke y S. Rohjans, «Evaluation of smart grid control strategies in co-simulation — integration of IPSYS and MOSAIK,» de *2014 Power Systems Computation Conference*, Wroclaw, Poland, IEEE, 2014, pp. 1-7.
- [75] W. Li, A. Monti, M. Luo y R. A. Dougal, «VPNET: A co-simulation framework for analyzing communication channel effects on power systems,» *2011 IEEE Electric Ship Technologies Symposium*, pp. 143-149, 2011.
- [76] W. Li, M. Luo, L. Zhu, A. Monti y F. Ponci, «A co-simulation method as an enabler for joint analysis and design of MAS-based electrical power protection and communication,» *SIMULATION*, vol. 89, n° 7, pp. 790-809, Mar 2013.
- [77] W. Li, M. Ferdowsi, M. Stevic, A. Monti y F. Ponci, «Cosimulation for Smart Grid Communications,» *IEEE Transactions on Industrial Informatics*, vol. 10, n° 4, pp. 2374-2384, 2014.
- [78] A. K. Bharati y V. Ajjarapu, «SMTD Co-Simulation Framework With HELICS for Future-Grid Analysis and Synthetic Measurement-Data Generation,» *IEEE Transactions on Industry Applications*, vol. 58, n° 1, pp. 131-141, Oct 2022.
- [79] B. Sergi y K. Pambour, «An Evaluation of Co-Simulation for Modeling Coupled Natural Gas and Electricity Networks,» *Energies*, vol. 15, n° 14, 2022.
- [80] N. Panossian, T. Elgindy, B. Palmintier y D. Wallison, «Synthetic, Realistic Transmission and Distribution Co-Simulation for Voltage Control Benchmarking,» *2021 IEEE Texas Power and Energy Conference (TPEC)*, pp. 1-5, Feb 2021.
- [81] W. Wang, X. Fang, H. Cui, F. Li, Y. Liu y T. J. Overbye, «Transmission-and-Distribution Dynamic Co-Simulation Framework for Distributed Energy Resource Frequency Response,» *IEEE Transactions on Smart Grid*, vol. 13, n° 1, pp. 482-495, 2022.
- [82] W. H. Kersting y G. Shirek, «Short circuit analysis of IEEE test feeders,» de *PES T&D 2012*, IEEE, 2012, pp. 1-9.
- [83] Grid Modernization Laboratory Consortium (GMLC-TDC), «HELICS-Tutorial,» GitHub, 2020. [En línea]. Available: <https://github.com/GMLC-TDC/HELICS-Tutorial-2020-03-13/tree/master/beginner>.
- [84] T. D. Hardy, B. Palmintier, P. L. Top, D. Krishnamurthy y J. C. Fuller, «Co-simulation Stages - HELICS documentation,» 2021. [En línea]. Available: <https://docs.helics.org/en/latest/user->

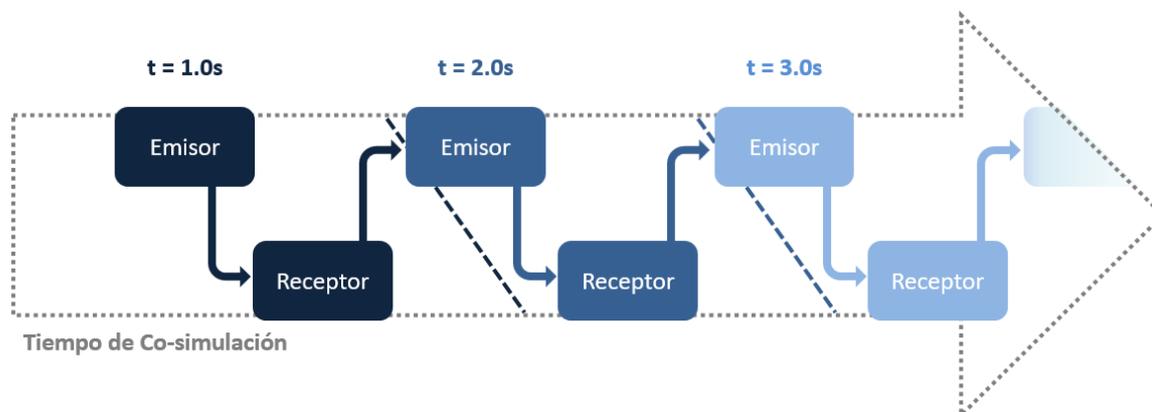
guide/fundamental\_topics/stages.html.

- [85] ESS, «¿Cuál es la Importancia de un Ingeniero en la Simulación Computacional? - ESSS,» 2021. [En línea]. Available: <https://www.esss.com/es/blog/cual-es-la-importancia-de-un-ingeniero-en-la-simulacion-computacional/>.



# ANEXO A. EJEMPLO DE CO-SIMULACIÓN SIMPLE UTILIZANDO HELICS

A continuación, se presenta un ejemplo de una co-simulación de un sistema distribuido donde dos entidades intercambian información de manera sincronizada y controlada temporalmente. La finalidad de este ejemplo es mostrar cómo se ejecuta y todas las partes que intervienen para realizar una co-simulación de forma eficaz. Las entidades son ejecutadas con federados individuales y la sincronización temporal es mantenida por un ‘broker’ de HELICS. En el marco de co-simulación, el intercambio de información entre simuladores en términos de tiempo de simulación se configura como débilmente acoplado, como se muestra en la **Figura A-1**. Nótese que el intercambio de información es sincronizado por HELICS. Se puede apreciar una flecha discontinua que apunta hacia la derecha, esta indica el tiempo de simulación; y las líneas diagonales discontinuas separan los estados cambiantes (en términos de tiempo de simulación) de las entidades a través del tiempo de co-simulación [83].



**Figura A-1.** Flujo de simulación en términos del tiempo de simulación del ejemplo. Elaboración propia.

Como se puede apreciar en la figura anterior, dentro de los elementos que componen esta co-simulación de ejemplo tenemos las dos entidades que son ‘**Emisor\_A**’ y ‘**Receptor\_A**’, quienes representan dos federados que se comunican de forma eficaz; pero también tenemos al ‘**Runner\_A**’ que es el elemento de configuración y quien hace ejecutar la co-simulación entre las dos entidades mediante HELICS.

El ‘**Emisor\_A**’ representa un federado que envía datos a un tema específico (llamado ‘tema\_A’) en intervalos de tiempo definidos. Este federado realiza cálculos basados en la constante pi multiplicada por el tiempo en cada iteración de su bucle de simulación y publica estos valores en el tema designado.

El Código A.2 consiste en el archivo de configuración, el cual define las características y comportamiento del emisor en un sistema de comunicación distribuida. Indica el tipo de núcleo o motor que se utiliza para la comunicación (en este caso, se está utilizando “zmq” o ZeroMQ, que es una biblioteca de mensajería asíncrona); establece el nivel de detalle de los registros (en este caso, el nivel es "summary", lo que significa que solo se registrarán mensajes resumidos); también indica qué tipo de datos espera recibir, la frecuencia de actualización, y otros detalles relacionados con la configuración y el comportamiento del emisor en la red. Cada federado debe tener su propio archivo de configuración como ya se ha mencionado anteriormente.

**Código A.1** Script ‘Emisor\_A’ para el ejemplo.

```

import helics as h
from math import pi

# ----- Etapa 1 - Creación -----
fed_A = h.helicsCreateCombinationFederateFromConfig("Emisor_A.json")
print(f"{fed.name}: Combination federate created")

# ----- Etapa 2 - Inicialización -----
tema_A = h.helicsFederateGetPublication(fed_A, "tema_A")
print(f"{fed.name}: Publication registered")

# ----- Etapa 3 - Ejecución -----
h.helicsFederateEnterExecutingMode(fed_A)
print(f"{fed.name}: Entering execution mode")

current_time = 0

for t in range(0, 10):
    test_A = round(pi*t, 5);
    while current_time < t:
        current_time = h.helicsFederateRequestTime(fed_A, t)
        print(f"Sending value: a = {test_A}, at time: t = {current_time}")
        h.helicsPublicationPublishDouble(tema_A, test_A)

# ----- Etapa 4 - Estado Final -----
h.helicsFederateDisconnect(fed_A)
print(f"{fed.name}: Federate finalized")
h.helicsFederateFree(fed_A)
h.helicsCloseLibrary()

```

**Código A.2** Archivo de configuración ‘Emisor\_A’ para el ejemplo.

```

{
  "coreType": "zmq",
  "loglevel": "summary",
  "name": "Emisor_A",
  "publications": [
    {
      "global": true,
      "key": "tema_A",
      "required": true,
      "type": "double"
    }
  ],
  "timeDelta": 1.0
}

```

Por otro lado, el ‘**Receptor\_A**’ simula un federado receptor que, a través de la biblioteca HELICS, se suscribe al tema ‘tema\_A’ y recibe los datos enviados por el federado ‘**Emisor\_A**’. Este federado avanza en el tiempo simulado de manera sincronizada con el emisor, obteniendo los valores publicados en el tema en cada intervalo de tiempo y mostrando dichos valores junto con el tiempo correspondiente.

**Código A.3** Script ‘Receptor\_A’ para el ejemplo.

```

import helics as h

# ----- Etapa 1 - Creación -----
fed_A = h.helicsCreateCombinationFederateFromConfig("Receptor_A.json")
print(f"{fed.name}: Combination federate created")

# ----- Etapa 2 - Inicialización -----
tema_A = h.helicsFederateGetInputByTarget(fed_A, "tema_A")
print(f"{fed.name}: Subscription registered")

# ----- Etapa 3 - Ejecución -----
h.helicsFederateEnterExecutingMode(fed_A)
print(f"{fed.name}: Entering execution mode")

current_time = 0

for t in range(1, 11):
    while current_time < t:
        current_time = h.helicsFederateRequestTime(fed_A, t)
        input_A = h.helicsInputGetDouble(tema_A)
        print(f"Received value: a = {input_A}, at time: t = {current_time}")

# ----- Etapa 4 - Estado Final -----
h.helicsFederateDisconnect(fed_A)
print(f"{fed.name}: Federate finalized")
h.helicsFederateFree(fed_A)
h.helicsCloseLibrary()

```

**Código A.4** Archivo de configuración ‘Receptor\_A’ para el ejemplo.

```

{
  "coreType": "zmq",
  "loglevel": "summary",
  "name": "Receptor_A",
  "subscriptions": [
    {
      "global": true,
      "key": "tema_A",
      "required": true,
      "type": "double"
    }
  ],
  "timeDelta": 1.0
}

```

El archivo de configuración ‘**Runner\_A**’ actúa como un planificador para la ejecución de estos federados. Define dos federados, ‘**Emisor\_A**’ y ‘**Receptor\_A**’, que se ejecutan localmente y se comunican a través del intermediario HELICS, configurado como un ‘**broker**’, quien facilita la comunicación entre federados. Además, establece los comandos de ejecución para cada federado, especificando los scripts de Python que se utilizarán para cada uno de ellos.

Código A.5 Archivo de configuración 'Runner\_A' para el ejemplo.

```

{
  "name": "Simple",
  "broker": true,
  "federates": [
    {
      "directory": ".",
      "exec": "python emisor.py 1",
      "host": "localhost",
      "name": "Emisor"
    },
    {
      "directory": ".",
      "exec": "python receptor.py 1",
      "host": "localhost",
      "name": "Receptor"
    }
  ]
}

```

En la **Figura A-2** se muestra cómo se vería el proceso al ejecutar el comando para iniciar la co-simulación de este ejemplo. Se puede apreciar como HELICS crea el 'broker' para la co-simulación y también como hace interactuar los federados creados y descritos anteriormente, con esto se logra de forma efectiva la co-simulación descrita en este ejemplo. Por otro lado, en la **Figura A-3** se puede apreciar la estructura del directorio utilizada para realizar este ejemplo; mientras más compleja sea la co-simulación a realizar más complejo será su directorio.

```

(helics) C:\Users\ElvisMarmol\Documents\US MSEE\00 TFM\Ejemplo simple>helics run --path=Runner_A.json
[info] Running federation: Simple_A
[info] Adding auto broker (i.e. 'helics_broker -f2') to helics-cli subprocesses.
[info] Running federate Emisor as a background process
[info] Running federate Receptor as a background process
[info] Running federate broker as a background process
[info] Waiting for 3 processes to finish ...
[info] Done.

```

Figura A-2. Ventana de comando de ejecución de la co-simulación del ejemplo. Elaboración propia.

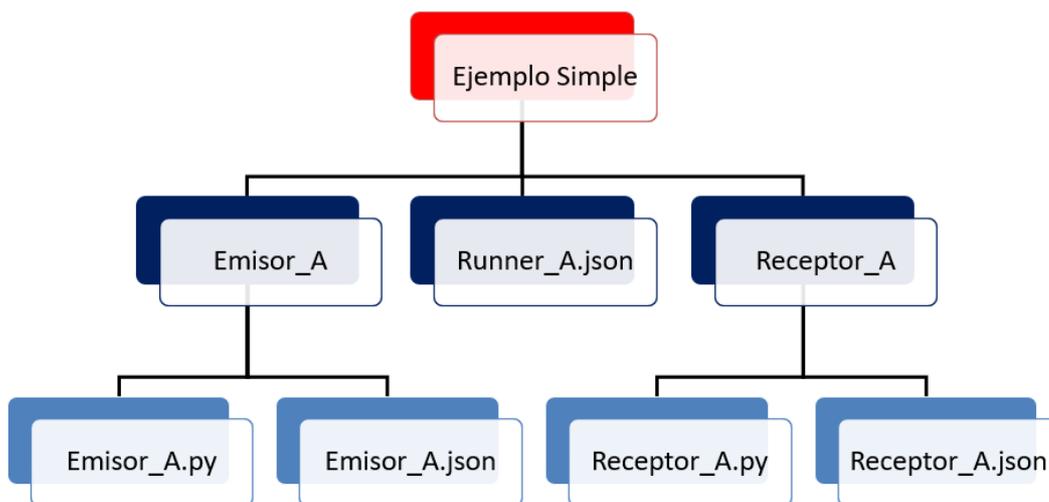


Figura A-3. Estructura del directorio de la co-simulación del ejemplo. Elaboración propia.

En esta co-simulación se observan los valores generados en los archivos de registro ('Emisor.log' y 'Receptor.log') que muestran la comunicación entre el emisor y el receptor en la co-simulación. Estos archivos no aparecen en la estructura del directorio porque solo se toman en cuenta los archivos necesarios para la ejecución de la co-simulación.

En el archivo '**Emisor.log**' se encuentra registrada la secuencia de valores enviados con sus respectivos tiempos de envío. Cada línea muestra el valor generado por el emisor y el momento exacto en el que fue enviado. Estos valores siguen una secuencia basada en la fórmula utilizada por el emisor para calcular los datos, multiplicando la constante pi por los intervalos de tiempo del 1 al 10.

#### Código A.6 Registro del federado 'Emisor\_A' para el ejemplo.

```
Emisor_A: Combination federate created
Emisor_A: Publication registered
Emisor_A: Entering execution mode
Sending value: a = 3.14159, at time: t = 1.0
Sending value: a = 6.28319, at time: t = 2.0
Sending value: a = 9.42478, at time: t = 3.0
Sending value: a = 12.56637, at time: t = 4.0
Sending value: a = 15.70796, at time: t = 5.0
Sending value: a = 18.84956, at time: t = 6.0
Sending value: a = 21.99115, at time: t = 7.0
Sending value: a = 25.13274, at time: t = 8.0
Sending value: a = 28.27433, at time: t = 9.0
Emisor_A: Federate finalized
```

Por otro lado, en el archivo '**Receptor.log**', se presentan los valores recibidos por el federado receptor. Aquí, los valores iniciales parecen ser incorrectos o no inicializados, mostrando un valor atípico en la primera línea (0). Sin embargo, los valores correctos comienzan desde la segunda línea, reflejando los valores recibidos por el federado receptor en cada intervalo de tiempo. Los valores recibidos coinciden con los valores enviados por el emisor, pero con un desfase de un intervalo de tiempo, lo que indica que el receptor recibe los datos del emisor un paso de tiempo después de que estos se envían.

#### Código A.7 Registro del federado 'Receptor\_A' para el ejemplo.

```
Receptor_A: Combination federate created
Receptor_A: Subscription registered
Receptor_A: Entering execution mode
Received value: a = 0.0, at time: t = 1.0
Received value: a = 3.14159, at time: t = 2.0
Received value: a = 6.28319, at time: t = 3.0
Received value: a = 9.42478, at time: t = 4.0
Received value: a = 12.56637, at time: t = 5.0
Received value: a = 15.70796, at time: t = 6.0
Received value: a = 18.84956, at time: t = 7.0
Received value: a = 21.99115, at time: t = 8.0
Received value: a = 25.13274, at time: t = 9.0
Received value: a = 28.27433, at time: t = 10.0
Receptor_A: Federate finalized
```

Esta discrepancia inicial (0) en la recepción puede indicar un problema de sincronización o inicialización del receptor al inicio de la simulación. A pesar de este desfase inicial, los valores recibidos después de la primera

línea siguen un patrón coherente y son consistentes con los valores enviados por el emisor, aunque se reciben un intervalo de tiempo después de ser emitidos.

Los resultados obtenidos de esta co-simulación son significativos, ya que validan la capacidad de HELICS para facilitar la interacción entre componentes simulados, permitiendo la transferencia efectiva de datos entre entidades distribuidas. Además, demuestran la sincronización exitosa entre el emisor y el receptor, lo que subraya la utilidad de esta herramienta para simular entornos complejos donde la comunicación y el intercambio de información son fundamentales.

Este ejemplo de co-simulación ha evidenciado de forma simple la viabilidad y eficacia de utilizar la HELICS para la comunicación entre componentes en un entorno simulado, proporcionando una base sólida para explorar y comprender mejor los sistemas distribuidos y las interacciones entre entidades en futuras investigaciones y aplicaciones prácticas.

