

# Aproximación metodológica a la implantación del gemelo digital en buques

Autora: Sara Ferreño González

---

Tesis doctoral UDC / 2022

Directores: Vicente Díaz Casás  
Marcos Míguez González

Tutor: Vicente Díaz Casás

Programa Oficial de Doctorado en  
Ingeniería Naval e Industrial



UNIVERSIDADE DA CORUÑA



A mis padres y hermanos



# RESUMEN

---

En el camino de la transformación del sector marítimo hacia lo que se ha denominado *Industria 4.0* se puede identificar un elemento clave: el gemelo digital. Un gemelo digital es un modelo virtual de un producto o un proceso, con el cual mantiene un constante intercambio de información y que le permite replicar el comportamiento del sistema real, adaptarse a los cambios, y proponer mejoras en él.

El gemelo digital del buque conforma un entorno digital que aglutinará toda la información existente (tanto de diseño como de operación) y la dotará de coherencia y estructura para conformar un conjunto con capacidades superiores a la agregación de cada parte individual.

En esta tesis doctoral se propone una metodología para el desarrollo e implantación del gemelo digital del buque. Se propondrá la aplicación práctica a sistemas reales del buque, abordando aspectos como: la adquisición de datos desde el sistema real, el desarrollo de modelos de simulación de los sistemas o el uso de algoritmos basados en inteligencia artificial que permitan avanzar en el camino hacia un gemelo digital inteligente.



# RESUMO

---

No camiño da transformación dixital do sector marítimo cara o que se denominou como *Industria 4.0* pódese identificar un elemento chave: o xemelgo dixital. Un xemelgo dixital é un modelo virtual dun produto ou dun proceso, co cal mantén un constante intercambio de información e que permite replicar o comportamento do sistema real, adaptarse aos cambios e propor melloras nel.

O xemelgo dixital do buque conforma un entorno dixital que reúne toda a información existente (tanto de deseño coma de operación) e dótaa de coherencia e estrutura para conformar un conxunto con capacidades superiores á agregación de cada unha das partes de xeito individual.

Nesta tese de doutoramento proponse unha metodoloxía para o desenvolvemento e a implantación do xemelgo dixital do buque. Proporase a aplicación práctica a sistemas reais do buque, abordando aspectos como: a adquisición de datos dende o sistema físico, o desenvolvemento de modelos de simulación dos sistemas ou o uso de algoritmos baseados en intelixencia artificial que permitan avanzar no camiño cara un xemelgo dixital intelixente.





# ABSTRACT

---

On the path of transformation of the maritime sector towards Industry 4.0, a key element can be identified: the digital twin. A digital twin is a virtual model of a product or a process. The virtual model maintains a constant exchange of information with the real asset and allows the model to replicate the behavior of the real system, adapt to changes, and propose improvements to it.

The digital twin of a ship creates a digital environment that will bring together all the existing information (both design and operation) and will provide it with coherence and structure to set up greater capabilities than the aggregation of each individual part.

The main goal of this thesis is to propose a methodology for the development and implementation of the digital twin of a ship. A practical application to real ship systems will be proposed, addressing aspects such as: data acquisition from the real system, development of simulation models and the use of algorithms based on artificial intelligence that allow new improvements in the path to a smart digital twin.



# AGRADECIMIENTOS

---

En primer lugar, me gustaría dar las gracias a mis directores de tesis, los Profesores Vicente Díaz y Marcos Míguez, por su gran apoyo y su paciencia. En segundo lugar, a mis compañeros en la Unidad Mixta de Investigación: Alicia, Pedro, Jaouad y Daniel. Gracias también a los directores de la UMI (sobre todo a los Profesores Daniel Pena y Pablo Fariñas), y a los responsables (por parte de Navantia y de la UDC) de las actuaciones en las que tuve el placer de participar durante estos años. A Navantia y a la Armada Española por facilitarme la realización de pruebas en sus instalaciones y buques.

Gracias a toda la gente del GII, por crear un ambiente de trabajo inmejorable, y en especial al grupito del café, que son mi segunda familia.

Gracias a mis padres, Pura y Juan, y a mis hermanos, Juanjo y Roberto. Sin ellos a mi lado, esto no habría sido posible. Y gracias, también, a Luis que me ha aguantado en este largo camino; y al que le debo muchas tardes de sol por todas las que nos hemos perdido durante este tiempo.

Gracias a todos mis amigos, a los que he visto poco durante este tiempo.

Y gracias a todos los que alguna vez me han preguntado: “Y la tesis, ¿qué?”.



## PUBLICACIONES

---

- Díaz-Casas, V., Vilar-Montesinos, M., Munín-Doce, A., Díaz-Bouza, M.-A., Trueba-Martínez, P., & **Ferreno-Gonzalez, S.** (2018). Industrial Internet of Things (IIOT): The digital challenge of Shipyard 4.0. *Ingeniería Naval*, 965.
- Díaz-Casas, V., Doce, A. M., Martínez, P. T., **Ferreno-Gonzalez, S.**, & Vilar, M. (2018). Shipyards of the 21st century: Industrial internet of things on site. *Progress in Maritime Technology and Engineering - Proceedings of the 4th International Conference on Maritime Technology and Engineering, MARTECH 2018*, 439–446. <https://doi.org/10.1201/9780429505294-50>
- Munín-Doce, A., Trueba-Martínez, P., **Ferreno-Gonzalez, S.**, Díaz-Casas, V., Vilar-Montesinos, M., & Ferreiroa-García, R. (2019). Industrial Internet of Things: aplicación práctica a un taller. *Ingeniería Naval*, 986.
- Trueba-Martínez, P., **Ferreno-Gonzalez, S.**, Munín-Doce, A., García, C., & Parente, S. (2020). Análisis de la implementación de un gemelo digital a bordo de un buque. *Ingeniería Naval*, 988.
- Ferreno-Gonzalez, S.**, Díaz-Casas, V., & García-San-Gabino, C. (2020). Modelo virtual del sistema de CI para detección y respuesta autónoma ante averías. *Ingeniería Naval*, 988.
- Munín-Doce, A., Díaz-Casas, V., Trueba, P., **Ferreno-González, S.**, & Vilar-Montesinos, M. (2020). Industrial Internet of Things in the production environment of a Shipyard 4.0. *International Journal of Advanced Manufacturing Technology*, 108(1–2), 47–59. <https://doi.org/10.1007/s00170-020-05229-6>
- Ferreno-Gonzalez, S.**, Díaz-Casas, V., Miguez-Gonzalez, M., & Garcia-San-Gabino, C. (2021). Virtual model of FiFi system for autonomous detection and response to failure. In *Developments in Maritime Technology and Engineering* (pp. 231–237). <https://doi.org/10.1201/9781003216582-25>



# 1 CONTENIDO

---

1.	Introducción .....	5
1.1.	Objetivos .....	5
1.2.	Metodología .....	6
2.	Antecedentes .....	11
1.1	Industria 4.0.....	11
1.1.1	Iniciativas Industria 4.0.....	15
1.2	Industria Naval y Marítima 4.0 .....	17
1.2.1	Iniciativas Industria Naval y Marítima 4.0.....	19
1.3	Tecnologías 4.0.....	25
1.4	Tecnologías 4.0 para gemelo digital.....	29
1.4.1	Internet de las Cosas en la Industria. ....	30
1.4.2	Simulación.....	32
1.4.3	Inteligencia Artificial .....	33
1.4.4	Analítica de Datos y Big Data .....	35
1.5	Gemelo Digital .....	36
1.5.1	El concepto de gemelo digital .....	37
1.6	Gemelo digital en la construcción naval e industria marítima.....	38
1.6.1	Algunas aplicaciones reales.....	41
1.7	Gemelo digital en la construcción naval militar .....	42
3.	Desarrollo metodológico para implantación del GD .....	47
3.1.	Definición de gemelo digital.....	47
3.2.	Etapas para la creación de un gemelo digital.....	52
3.3.	Características intrínsecas de un gemelo digital.....	54

3.4. Requisitos de implementación de un gemelo digital.....	58
3.5. Arquitectura general para implementación de un Gemelo Digital	59
3.5.1. Sistema físico .....	61
3.5.2. Interfaz mundo físico / mundo virtual .....	63
3.5.3. Gemelo digital .....	65
3.6. Gemelo Digital en buques.....	74
3.6.1. Sistema físico .....	75
3.6.2. Interfaz mundo físico / mundo virtual .....	76
3.6.3. Gemelo Digital .....	77
3.6.4. Otros aspectos relevantes en la implantación del GD del buque. ¿Gemelo digital del buque embarcado o en tierra? .....	85
3.7. Plataformas de implementación del Gemelo Digital .....	87
4. Implementación del gemelo digital .....	89
4.1. Descripción del caso de uso.....	89
4.2. Sistema físico.....	90
4.2.1. Descripción del sistema .....	90
4.3. Descripción de las Arquitecturas planteadas para el despliegue del gemelo digital.....	93
4.4. Despliegue centralizado en plataforma IOT .....	95
4.4.1. Modelo de Datos y desarrollo de diagramas conceptuales.....	97
4.4.2. Adquisición de datos de campo .....	99
4.4.3. Interfaz campo y gemelo digital .....	100
4.4.4. Modelos de simulación .....	102
4.4.5. Modelos ejecutándose en la capa de gemelo digital.....	109
4.4.6. Almacenamiento de datos .....	114
4.4.7. Conexión con sistemas de información .....	115
4.4.8. Conclusiones.....	118
4.5. Arquitectura descentralizada .....	119



4.5.1.	Adquisición de datos hacia el gemelo digital .....	120
4.5.2.	Modelos de Simulación .....	121
4.5.3.	Conexión de los modelos de simulación con el gemelo digital. 140	
4.5.4.	Caso de uso. Detección de fallos en tiempo real .....	147
4.5.5.	Conclusiones .....	150
4.6.	Análisis de compatibilidad e integración de modelos de simulación FMU 152	
4.6.1.	Análisis de compatibilidad de modelos FMU .....	153
4.6.2.	Integración de modelos FMU .....	155
5.	Aplicación IA al gemelo digital .....	167
5.1.	Introducción .....	167
5.2.	detección de fallos en tuberías .....	167
5.2.1.	Detección de fallos en tuberías basada en inteligencia artificial 169	
5.3.	Conceptos básicos sobre redes neuronales .....	171
5.3.1.	La Neurona. Perceptrón .....	171
5.3.2.	Red neuronal .....	172
5.3.3.	Tipología de red seleccionada .....	174
5.3.4.	Entrenamiento de redes neuronales .....	174
5.4.	Descripción del Sistema empleado como caso de uso .....	177
5.5.	Modelos Numéricos. Simulaciones .....	179
5.5.1.	Descripción general del modelo para el sistema contraincendios 179	
5.6.	Pruebas en buque real. FASE 1 .....	182
5.6.1.	Preparación de las pruebas .....	183
5.6.2.	Realización de pruebas .....	184
5.6.3.	Conclusiones/resumen de las pruebas realizadas .....	187

5.6.4. Validación del modelo de simulación .....	188
5.6.5. Simulaciones realizadas para generación de datos.....	190
5.7. Desarrollo del sistema de detección de fallos .....	192
5.7.1. Diseño de la red .....	192
5.7.2. Entrenamiento de la red.....	195
5.7.3. Resultados.....	197
5.8. Pruebas en buque real. FASE 2.....	201
5.8.1. Preparación de las pruebas.....	202
5.8.2. Realización de pruebas .....	203
5.8.3. Conclusiones/resumen de las pruebas realizadas.....	206
5.9. Rediseño/Reentrenamiento del sistema de detección de fallos. Desarrollo del sistema de localización de fallos .....	207
5.9.1. Rediseño y reentrenamiento de la red neuronal para detección de fallos	207
5.9.2. Desarrollo de sistema para localización de fallos .....	214
5.10. Implementación de sistemas basados en RN en el buque.....	222
5.11. Conclusiones .....	225
6. Conclusiones y trabajo futuro.....	228
6.1. Implantación de la metodología propuesta en un entorno real ....	229
6.2. Desarrollo de algoritmos basados en inteligencia artificial en el gemelo digital para detección y localización de fallos en el buque .....	231
7. Referencias.....	233
8. Apéndice A.....	254
9. Listado de figuras .....	257
10. Listado de Tablas.....	264

# 1. INTRODUCCIÓN

---

El concepto de gemelo digital está tomando cada vez más protagonismo en la industria. Se trata de una tecnología transversal y disruptiva en el desarrollo de lo que se ha denominado Industria 4.0. En resumen, un gemelo digital es una representación virtual de un activo físico con capacidad de representar el estado del activo en tiempo real, de modo que un gemelo digital puede ser empleado para evaluar el estado del activo, pero, además, permite predecir su comportamiento u optimizar su funcionamiento.

En todos los sectores de la industria se ha despertado un creciente interés hacia el desarrollo de gemelos digitales para sus productos y procesos. En algunos casos, ya se cuenta con desarrollos maduros, como puede ser el caso de la automoción o el sector aeronáutico. Sin embargo, se puede observar que existe una importante carencia relacionada con el desarrollo de gemelo digital en la industria marítima. Por este motivo, se decidió abordar en esta tesis doctoral aspectos relativos a la implantación del gemelo digital en el buque. Debido a que el buque es un sistema heterogéneo complejo, con sistemas de muy diferente naturaleza, y diferente nivel de sensorización, pueden existir importantes barreras tecnológicas para desarrollar su gemelo digital (pueden presentarse dificultades, por ejemplo, para la monitorización de señales, o para su almacenamiento) lo cual, unido a la complejidad tecnológica que implica tratar de avanzar un paso más respecto a representaciones digitales y hacia modelos (dinámicos y actualizables) que permitan simular fielmente un sistema, representa un importante reto.

## 1.1. OBJETIVOS

Los principales objetivos en el desarrollo de esta tesis son:

- Desarrollo de una metodología genérica sienta las bases para la implantación de un gemelo digital en buques.
- Validar la metodología propuesta mediante su aplicación a casos prácticos de uso.

Para alcanzar estos objetivos principales, se plantean los siguientes objetivos secundarios:

- Analizar diferentes alternativas para la arquitectura de gemelos digitales en buques.
- Experimentar en un entorno equivalente al de un buque real la adquisición de datos desde el sistema físico hacia el gemelo digital.
- Evaluar en un entorno real la mayor cantidad de componentes definidos en la arquitectura planteada para el gemelo digital.
- Desarrollar modelos de simulación para los sistemas del buque y su implantación y ejecución en el entorno del gemelo digital.
- Implantar algoritmos de inteligencia artificial en el gemelo digital de sistemas del buque.

## 1.2. METODOLOGÍA

Para satisfacer los objetivos planteados, se planteó la siguiente metodología:

- 1) En primer lugar, se realizará un análisis del estado actual de la tecnología y de su aplicación a buques que permita conocer en qué punto de maduración se encuentran las tecnologías relacionadas con el gemelo digital en general, y en concreto, con el gemelo digital del buque, así como aquellos puntos de mayor interés y aquellos que requieren mayor desarrollo.
- 2) En segundo lugar, se llevará a cabo el desarrollo teórico que defina las consideraciones que se deben de tener en cuenta a la hora de implementar un gemelo digital, y cuales deben de ser los pasos necesarios para realizar esa implementación. Para ello, se definirán los principales requisitos y características con las que debe contar un gemelo digital, así como una definición detallada de los componentes elementales que deben de formar parte de este y como estarán interconectados (definición de la arquitectura). Además, se detallarán aquellas consideraciones específicas que se deberán valorar y sobre las que recaerá el peso a la hora de plantear el gemelo digital de un buque.

3) En tercer lugar, la base teórica desarrollada en los puntos anteriores se validará empleando dos casos prácticos:

3.a) Un primer caso, centrado en la aplicación de la metodología planteada aplicada a un caso de uso y abarcando los siguientes aspectos:

- Adquisición de datos desde el sistema real hacia el gemelo digital.
- Desarrollo de modelos de simulación para ejecución en tiempo real.
- Almacenamiento de datos.
- Visualización de información en el entorno del gemelo digital.

3.b) Un segundo caso práctico, basado también en el punto 2 y complementario al 3.a), centrado en la aplicación de inteligencia artificial en a un sistema del buque, tratando de avanzar hacia el buque inteligente.

Siguiendo la metodología propuesta, esta tesis doctoral se ha estructurado en 6 capítulos. Además de un capítulo inicial introductorio y uno final en el que se detallan las conclusiones y el trabajo futuro, se presentan cuatro capítulos principales:

- El capítulo 2 se inicia con un estado del arte acerca de la Industria 4.0 en general y de la Industria 4.0 en el sector naval. A continuación, se analizarán los avances relacionados con las principales tecnologías habilitadoras, centrándose en aquellas que cobran especial relevancia en el desarrollo de gemelos digitales: simulación, Internet de las Cosas, analítica de datos e inteligencia artificial. Finalmente, se analizará el concepto de gemelo digital, sus orígenes, y los principales desarrollos relacionados con el gemelo digital del buque.
- En el capítulo 3 se planteará, en base a la bibliografía analizada, una metodología que permita desplegar la tecnología de gemelo

digital en un buque o, más en concreto, en los sistemas del buque. Se analizarán las principales características con las que debe contar un gemelo digital y se definirá su arquitectura estableciendo cuáles deben de ser los componentes elementales con los cuales debe contar y cómo estos deben de estar conectados entre sí. Finalmente, se concretará la aplicación de esa definición de arquitectura para la aplicación de la metodología propuesta a un buque y qué criterios concretos será necesario valorar en este caso.

- En el capítulo 4 se presentará el primer caso práctico desarrollado. Este caso práctico pretende aplicar los aspectos descritos en el capítulo 3 a un sistema real del buque, en este caso, un sistema HVAC diseñado para ser objeto de pruebas piloto relacionadas con gemelos digitales y otras tecnologías disruptivas de la Industria 4.0. Los trabajos realizados en este demostrador se desarrollaron en 2 fases, en las que se analizaron 2 alternativas diferentes para la arquitectura a desplegar en la implantación del gemelo digital del sistema HVAC: una primera fase en la que se evaluó una arquitectura basada, o centralizada en una plataforma comercial de IoT (Thingworx, de PTC). Y una segunda fase en la que se analizó la implantación de un gemelo, que, a diferencia de la primera, no se encontraba centralizado, sino que cada componente de la arquitectura puede estar alojado en diferentes entornos (por ejemplo, modelos ejecutándose en capa local).
- El capítulo 5 se centrará en el desarrollo de una prueba de concepto para aplicación de algoritmos basados en inteligencia artificial en el gemelo digital. Para ello, se seleccionó el sistema contra incendios por agua salada del buque para tratar de, empleando las señales de los sensores de presión con los que cuenta el sistema, desarrollar un modelo de machine learning que permita determinar si esos datos de presión se corresponden con un funcionamiento normal del sistema, o si, por el contrario, se trata de un fallo (rotura) de algún tramo de tuberías. Como complemento a lo anterior, también se tratará de determinar, empleando las medidas recogidas en todos los sensores, la localización de esa rotura.

Cabe destacar que parte del trabajo presentado en esta tesis ha sido desarrollado en el marco de la Unidad Mixta de Investigación, UMI, “El Astillero del Futuro” formada por la Universidade da Coruña (UDC), Navantia, y la Axencia Galega de Innovación (GAIN).

La UMI se constituyó como un consorcio para dar soporte a Navantia (e indirectamente a la industria auxiliar asociada) en su proceso de transformación hacia el marco de Industria 4.0. Las principales tecnologías facilitadoras aplicadas en esta transformación digital fueron:

- Internet de las cosas.
- Inteligencia artificial.
- Modelado y simulación.
- Realidad aumentada.
- Analítica de datos.
- Robótica y sistemas de asistencia a los operarios.

Dado que el foco principal de la UMI eran los procesos, inicialmente las líneas de investigación se orientaron a la optimización de los procesos en el astillero y a la implementación de las nuevas tecnologías de información y comunicaciones (TICs) en la construcción de buques.

La rápida evolución de la tecnología y con ella del mercado llevó, con posterioridad, a la necesidad de diversificar las actuaciones con algunas líneas de investigación orientadas también al producto, en concreto al por aquel entonces emergente programa de fragatas F110.





## 2. ANTECEDENTES

---

El gemelo digital, a menudo representado por el acrónimo GD (o DT, digital twin, en inglés), ha emergido como uno de los conceptos clave en el desarrollo de lo que se ha denominado Industria 4.0. Aunque se analizará en detalle más adelante, se puede decir que un Gemelo Digital es, en pocas palabras, un modelo virtual de un producto, un proceso o un servicio el cual se alimenta de la información obtenida de sensores o automatismos del sistema físico. Se trata de la unión entre el mundo físico y el mundo virtual, que en general nos va a permitir la monitorización de los sistemas, y a la vez el acceso a un análisis inteligente de los datos.

En este capítulo se analizarán los avances tecnológicos que han permitido llegar a plantear el desarrollo de gemelos digitales: qué significa ese nuevo paradigma, cuáles son las tecnologías en las que se basa y qué iniciativas relacionadas con industria 4.0 destacan en el mundo, y dentro de esas iniciativas cuáles hay relativas a la industria naval y marítima. La segunda parte del capítulo estará centrada en el concepto de gemelo digital, qué significa, qué componentes principales tiene y qué ventajas aporta su implementación, y cuáles son las tecnologías clave (de entre las KET) que se necesitan desplegar para el desarrollo de un gemelo.

### 1.1 INDUSTRIA 4.0

La Primera Revolución Industrial, que tuvo lugar en la segunda mitad del S XVIII en Gran Bretaña, supuso una importante transformación de la forma de producir. Surgió a partir del nacimiento de la máquina de vapor. Su introducción en la industria permitió aumentar de forma sustancial la capacidad de producción, pasando de la producción manual a la producción industrial. Además de ello, en esta época surgen las lámparas de gas, la locomotora y los barcos de vapor.

La Segunda Revolución Industrial se sitúa a finales del S XIX, caracterizada sobre todo por la aparición de nuevas fuentes de energía (electricidad y petróleo), el desarrollo de nuevos modos de producción en las fábricas y el desarrollo de las telecomunicaciones.

La aparición de la electricidad constituye el elemento clave de esta revolución industrial. El reemplazo del vapor por la electricidad supuso de nuevo importantes cambios en la forma de producir. Además, es en esta época cuando se inventa la bombilla, y surge con ella el alumbrado eléctrico. La aparición del petróleo permitió pasar de los motores de vapor a los de combustión interna, mejorando la propulsión de las locomotoras y de los barcos. Además, aparecieron nuevos medios de transporte: el automóvil y el avión. La aparición de la electricidad permitió el desarrollo de las telecomunicaciones: el telégrafo, el teléfono y la radio.

En cuanto a los avances relacionados con la mejora de la producción, en esta época surgen 2 conceptos clave: el taylorismo y el fordismo.

- Taylorismo: El concepto implicaba que un operario, en lugar de realizar muchas tareas diferentes, se especializase en una sola. Surge por lo tanto la producción en serie.
- Fordismo: O lo que es lo mismo, cadenas de montaje. Son las piezas las que se desplazan al operario.

Estos 2 conceptos, implicarán el desarrollo de la fabricación en masa y un importante aumento de la productividad.

Se puede considerar que la Tercera Revolución Industrial surgió a mediados del siglo XX, después de finalizar la segunda guerra mundial y se basó principalmente en el uso de nuevas tecnologías de la información y las comunicaciones y en las innovaciones que permitieron el desarrollo de energías renovables. El concepto de Tercera Revolución Industrial (o revolución científico-técnica o revolución de la inteligencia) fue introducido por el economista y sociólogo Jeremy Rifkin. (Rifkin, 2011). Esta Tercera Revolución tiene como ejes fundamentales el desarrollo de nuevas tecnologías de la información y comunicaciones que, junto al uso de las renovables, permitió llevar a cabo importantes cambios en muchas áreas: desarrollo de la informática y el despegue de internet, automatización de procesos (en este momento aparecen los primeros robots).

Estas son las 3 principales revoluciones industriales, que hasta ahora han tenido lugar. Si bien, desde principios de la década de los 2010s se ha empezado a hablar de que la sociedad se encuentra a las puertas de una 4ª

revolución industrial, a la que se bautizó como Industria 4.0. (Rajpurohit & Verma, 2016). Si en la primera revolución se puede destacar el paso de la producción manual a la mecanizada, en la segunda los principales hitos fueron la aparición de la electricidad y la fabricación en serie. En la tercera llegó la electrónica y las TIC y la 4ª revolución industrial trae consigo una tendencia clara hacia la total automatización de la fabricación basada principalmente, como se analizará más adelante, en la implantación de sistemas ciberfísicos y el Internet de las cosas (Kagermann et al., 2013).

El término Industria 4.0 apareció, como tal, por primera vez en 2011, cuando se presentó una iniciativa denominada “Industrie 4.0” por parte de una asociación en la que estaba representada el tejido empresarial, político y académico alemán, y mediante la que buscaban promover el concepto de Industria 4.0 con el objetivo de mejorar la competitividad de la industria alemana, convirtiéndose este país en el primero en introducir la Industria 4.0 en su agenda de gobierno, como una estrategia de alta tecnología (Hermann et al., 2015). En el año 2013 se publica un informe en el que sientan las bases de lo que implica este concepto, sus implicaciones y sus principales retos (Kagermann et al., 2013).

En general, se suelen citar como pilares fundamentales en los que se basa la industria 4.0 los siguientes (Hermann et al., 2015; Rajpurohit & Verma, 2016):

- Sistemas ciberfísicos (CPS). Se trata del eje central en torno al cual se vertebra este nuevo paradigma. Cuando se habla de sistemas ciberfísicos se hace referencia a objetos, dispositivos o equipamientos físicos embebidos, los cuales interactúan con el mundo virtual a través de una red de comunicaciones (Ferrer et al., 2018). La separación clásica entre sensor y actuador se está acortando cada vez más y se espera que desaparezca cuando el uso de sistemas ciberfísicos esté totalmente generalizado. El desarrollo de estos sistemas se puede desagregar en tres fases:
  - Una primera fase, en la que se desarrolla una primera generación de CPS que incluyen tecnologías de identificación como las etiquetas de identificación por radiofrecuencia, RFID, que permiten una identificación única. Además, el

- almacenamiento y la analítica se proporcionan como un servicio centralizado.
- La segunda generación de CPS está equipada con sensores y actuadores con una gama limitada de funciones.
  - Finalmente, los CPS de tercera generación pueden almacenar y analizar datos, están equipados con múltiples sensores y actuadores y son compatibles con la red.
- Internet de las cosas, (IoT): El término IoT se refiere a la interconexión digital entre dispositivos, personas y el propio internet, y el intercambio de información que se puede producir entre ellos. El despliegue del IoT ha sido la clave en el inicio del camino hacia la Industria 4.0 (Kalsoom et al., 2021). La integración del IoT en la fabricación permite la conexión en tiempo real de los sistemas ciberfísicos a la industria inteligente.
  - Internet de los servicios (IoS): Con este término se designan aquellos servicios asociados a un producto que pueden ser ofertados o adquiridos por una empresa a partir de las tecnologías TIC implantadas en esa empresa, y en especial, a gracias a las tecnologías IoT. (Reis & Gonçalves, 2018).
  - Fábrica inteligente (o tecnología inteligente, dependiendo de la bibliografía consultada) (Osterrieder et al., 2020). Una fábrica inteligente es aquella que es consciente de su contexto, y que además puede ayudar a las personas y las máquinas en la ejecución de sus tareas, basándose en información proveniente del mundo físico y virtual. La información del mundo físico podría ser, por ejemplo, posición o condición de una herramienta, y la información del mundo virtual podrían ser documentos electrónicos, planos o modelos de simulación.

M. Hermann cita como principios de diseño para la correcta implantación de la Industria 4.0, los siguientes (Hermann et al., 2015):

- Inter-operatividad: Habilidad de comunicarse entre los sistemas ciberfísicos y redes inteligentes.
- Virtualización: creación de una copia virtual de un mundo físico en forma de modelo virtual o modelo de simulación.

- Descentralización: capacidad para tomar sus propias decisiones sin depender de ningún elemento central.
- Capacidad “tiempo real”: En la industria 4.0 cada paso se identifica y registra en tiempo real. Aumenta la flexibilidad y ayuda en la personalización en tiempo real.
- Orientación al servicio: los servicios ya no deben formar parte de un sistema centralizado, sino que se ofrecen dentro de un sistema o fuera de las fronteras del sistema mediante Internet of Services IoS.
- Modularidad: el sistema completo debe constar de subsistemas más pequeños que se puedan reutilizar también para otros fines. Proporciona módulos individuales que se pueden actualizar fácilmente y tienen una capacidad de plug and play.

Para abordar el paradigma industrial que representa la Industria 4.0 será necesario aplicar una serie de tecnologías, lo que se ha denominado “Tecnologías Facilitadoras de la Industria 4.0”.

Más adelante se profundizará en las tecnologías facilitadoras, pero en general, cuando se habla de KETs se suelen citar las siguientes: big data, la robótica, la integración de los sistemas de información, el internet of things (IoT), la ciber-seguridad, la nube, la fabricación aditiva, la realidad aumentada y virtual, la simulación, vehículos autónomos, blockchain e inteligencia artificial.

### **1.1.1 Iniciativas Industria 4.0**

El Informe Capgemini y la consultora Deloitte, afirman que las fábricas inteligentes son la base de la Industria 4.0 (Deloitte, 2017; Rossmann et al., 2017). Alemania fue el primer país en crear una Hoja de Ruta relacionada con la Industria 4.0. En el año 2013 se elaboró el documento “Industrie 4.0” por parte del Ministerio de Educación e Investigación y el Ministerio de Asuntos Económicos y Energía (Kagermann et al., 2013).

En Estados Unidos se creó en el año 2014 el Industrial Internet Consortium (IIC), organización que busca la transformación de las empresas y de la sociedad mediante la aceleración del internet industrial de las cosas (IIoT) (Buchheit et al., 2020; Soley, 2015). Fue constituido por 5 empresas: AT&T, CISCO, GE, IBM e INTEL. Hoy en día cuenta con más de 150 miembros a lo largo de todo el mundo y están representadas entidades del ámbito industrial, académico y gubernamental. Trabajan en 19 grupos de trabajo relacionados con las

siguientes áreas principales: Transformación digital, Marketing, Seguridad y Tecnología.

El gobierno de China creó en el año 2015 una iniciativa a la que nombraron como “Made in China 2025” inspirada en la alemana “Industrie 4.0” y que propone una estrategia de reindustrialización basada en la fabricación inteligente como método para incrementar la competitividad, de manera que se pase de una era donde lo principalmente valorado era la cantidad a una nueva era donde la prioridad sea la calidad y eficiencia en la producción. Con este plan, China pretende ser líder en tecnología a escala internacional, por delante de potencias como Alemania, Estados Unidos o Japón (Institute for Security & Development Policy, 2018; Pérez-Cuadrado, 2016; Wübbecke et al., 2016).

En Japón, la Industria 4.0 se conoce como “Industrial Value Chain Initiative (IVI)” y surgió en el año 2015 después de un acuerdo de cooperación entre Japón y Alemania. Esta iniciativa tiene como objetivo final diseñar una nueva sociedad en la que las fábricas y las tecnologías de la información estén completamente integradas, haciendo hincapié en el capital humano, que a menudo queda postergado a un segundo plano. Se prestará especial atención a como afectarán los nuevos métodos de fabricación y la digitalización de la industria los humanos (Nishioka, 2016; Value & Initiative, 2019).

En Corea, la iniciativa de Industria 4.0 también fue renombrada: “Manufacturing Innovation 3.0 Strategy” o “Strategy 3.0”. Publicada en el año 2015, en esta Estrategia 3.0, la fábrica inteligente se define como un “sistema de fabricación en el que los procesos de planificación, diseño, producción, distribución y ventas están automatizados, conectados e integrados gracias a las tecnologías de la información, TICs”. (Wiktorsson et al., 2018). En Australia, la estrategia Industria 4.0 nació de nuevo a través de una colaboración con Alemania, en el año 2015 (Australia Germany Advisory Group, 2015) (Aljukic, 2017).

En España, la iniciativa Industria Conectada 4.0 (*Estrategias Para El Fomento de La Introducción*, 2020), es una iniciativa del gobierno español a través del Ministerio de Industria, que busca impulsar la transformación digital de la industria mediante la actuación conjunta y coordinada del sector público y privado (Fundación Telefónica, 2015).

Las principales líneas de actuación son las siguientes:

- Concienciación y formación.
- Entornos colaborativos y plataformas.
- Impulso a habilitadores digitales.
- Apoyo a la evolución digital de la industria.

## 1.2 INDUSTRIA NAVAL Y MARÍTIMA 4.0

Los nuevos paradigmas de la Industria 4.0 están motivando la realización de profundos cambios en todos los sectores industriales a nivel de transformación digital, con el principal objetivo de alcanzar una posición más competitiva en un mundo cada vez más globalizado. El sector marítimo y naval no pueden permanecer ajenos a este cambio de tendencia y debe de buscar su camino para enfrentarse a la nueva era digital.

En los últimos años se han producido importantes transformaciones en el sector de la construcción naval, que han implicado la implantación de:

- Nuevas máquinas.
- Nuevos procesos productivos.
- Nuevo software.
- Nuevas estructuras organizativas.

Si bien, en los próximos años se espera que sea un período de crecimiento, en el que habrá que enfrentarse principalmente a los siguientes retos (Stanic et al., 2018):

- Alcanzar una producción más eficiente.
- Alcanzar un ahorro de costes, eficiencia energética y protección medioambiental.
- Maximizar la seguridad del buque.

Todos los actores implicados están trabajando para satisfacer las demandas del mercado, mientras que al mismo tiempo aumenta la presión sobre la industria para mejorar la eficiencia energética. La Organización Marítima Internacional (OMI) se ha fijado como objetivo reducir la intensidad media de carbono (emisiones de CO<sub>2</sub> en transporte marítimo) en un 40% en 2030 y en un 70% en 2050 en comparación con los niveles de 2008. La OMI también tiene como objetivo reducir las emisiones totales de GEI (gases de efecto

invernadero) del transporte marítimo en al menos un 50% en 2050 en comparación con 2008. (International Maritime Organization, 2019)

En el informe McKinsey (World Economic Forum, 2018) se habla que la implantación de la Industria 4.0 tiene un potencial de creación de valor estimado en 2025 de 3.7 miles de millones de dólares. Aunque en 2018 apenas el 30% de las compañías estaban capturando valor a partir de las soluciones que aporta la Industria 4.0, cuando se estima que las industrias que invierten en digitalización podrían reducir sus costos operativos en un 3,6% y mejorar la eficiencia en un 4,1%. McKinsey también estima se pueden llegar a alcanzar mejoras en productividad del 3 al 5% mediante la implantación de Internet de las cosas, el consumo inteligente de energía y la supervisión y el control remotos. Además de ello el mantenimiento predictivo y el autoservicio guiado virtualmente también podrían reducir los costos de mantenimiento entre un 10% y un 40%.

El camino hacia la industria 4.0 está definido, y en algunos sectores que ya cuentan con un elevado grado de automatización y digitalización este camino no es muy largo. Pero no es así en el caso de la industria naval y marítima que en general presenta un grado de automatización y digitalización menor que otros sectores industriales, como puede ser el caso de la automoción, en la que las factorías cuentan con un elevado grado de automatización de tareas o de los edificios inteligentes, en los que se puede encontrar un avanzado grado de implantación de tecnologías relacionadas con IoT.

En el caso del sector naval, con la implantación de la Industria 4.0 se busca transformar desde la fase de diseño, construcción, operación, servicio y mantenimiento del buque, así como los sistemas de producción, y toda la cadena de valor.

Como se ha comentado, la implantación de la Industria 4.0 en cualquier sector, pasa por la implantación de las denominadas Tecnologías Habilitadoras, KET. Parece razonable pensar que las tecnologías habilitadoras definidas para la Industria 4.0 pueden aplicarse en su totalidad al sector de la construcción naval o a mundo marítimo. Sin embargo, algunas se están incorporando más rápidamente que otras; es el caso del internet de las cosas, simulación, impresión 3D o la robotización. Hay otras que en un futuro inmediato estarán incorporadas en el proceso constructivo de un astillero como son gemelo



digital, realidad aumentada, big data y cloud computing o la ciberseguridad. Por lo tanto, hoy en día, y sobre todo realizando una comparación respecto a otros sectores tradicionalmente más tecnológicos, como el sector de la automoción o el aeronáutico, es evidente que existe un déficit en la madurez digital de la industria de la construcción y la reparación naval.

Estas tecnologías pueden resultar bastante disruptivas en el sector naval, pero su aplicación cobra fuerza cada vez más debido a las ventajas que suponen en cuanto a optimización y eficiencia.

En el informe realizado por Interreg Atlantic Area en el marco del proyecto IN 4.0, (Interreg-Atlantic-Area, 2019) se lleva a cabo una estimación del grado de implementación de las diferentes tecnologías habilitadoras en el sector naval. Tecnologías como la Realidad Virtual y Aumentada, o la simulación de procesos está parcialmente implementada en algunos casos. Si bien, la mayoría de estas tecnologías solo cuentan con el desarrollo de experimentos piloto.

## **1.2.1 Iniciativas Industria Naval y Marítima 4.0**

Los principales países del mundo han abierto iniciativas orientadas a fomentar e implementar la industria 4.0 en las que en mayor o menor medida se contemplan medidas relacionadas con la construcción naval (Stanic et al., 2018). Algunas de estas iniciativas se describen a continuación:

### **1.2.1.1 Europa**

A nivel europeo, existen importantes iniciativas entre las que cabe destacar: KETmaritime “Transfer of Key Enabling Technologies (KETs) to the Maritime Industries” y USWE “Upskilling Shipbuilding Workforce in Europe”.

El proyecto KETmaritime busca promover las tecnologías habilitadoras clave (KETs): nanotecnología, biotecnología, materiales avanzados, fotónica y fabricación avanzada. La fabricación avanzada busca la aplicación de los últimos avances tecnológicos del sector productivo, con el objetivo de alcanzar optimización en los procesos y una mejora de la productividad, reducción de costes, mejora energética y medioambiental y también en la calidad de los productos. Estos avances tecnológicos contemplan: el escaneado, diseño e impresión 3D, la visión artificial, la realidad virtual y aumentada, la robótica flexible y colaborativa, la simulación de procesos productivos, las tecnologías de la comunicación avanzadas, etc. Como industria manufacturera, la construcción

naval puede beneficiarse de la implementación de muchas de estas tecnologías, promoviendo un cambio tecnológico que pueda incrementar la competitividad de las organizaciones que trabajan en el sector. (KETmaritime, 2019).

Actualmente, la fabricación avanzada se asocia con la implementación de tecnologías y mejoras como:

- Tecnologías para la fabricación digital: herramientas para diseño, simulación, ingeniería, fabricación, escaneado e impresión 3D.
- Infraestructuras, tecnologías y servicios TIC: redes inalámbricas, dispositivos móviles, cloud computing, hardware de desarrollo de bajo coste, wearables, etc.
- Automatización: IA (Inteligencia Artificial), redes neuronales, robótica colaborativa, visión artificial, robots y vehículos autónomos, etc.
- Tecnologías de simulación e inmersión: realidad virtual y realidad aumentada.

El proyecto USWE (USWE, 2020; Forecasting Trends & Challenges for a 4.0 Shipbuilding Workforce in Europe, 2020) nace con el objetivo de apoyar a la industria y los trabajadores en la adaptación a las tecnologías de la industria 4.0. Este proyecto reúne a representantes de la industria y de los trabajadores de toda Europa, así como expertos en educación para analizar el impacto del cambio tecnológico en los trabajos existentes y las necesidades de formación y empleo futuras. Uno de los paquetes de trabajo de este proyecto se centra en analizar el impacto de las tecnologías de la Industria 4.0 en el sector y el impacto potencial en las necesidades de habilidades y cualificaciones profesionales. En este caso se vuelven a abarcar tecnologías como el Big Data, Internet de las Cosas, Ciberseguridad, Robótica, Fabricación Aditiva o la Impresión 3D, pero desde un punto de vista de que su implantación en el sector naval tendrá un impacto significativo en las operaciones de los buques, en la mejora de la seguridad y el bienestar de los trabajadores.

En Alemania, se estima que se realizarán inversiones por un valor de 12.000 millones de euros en innovación para el despegue del Shipbuilding 4.0. Según la "Agenda Marítima 2025" (BWMi, 2017) y el "Estudio de Competencias Industriales y Tecnológicas en el Sector Naval", (European Defence Agency, 2020), la industria marítima alemana tiene una facturación anual de 18 mil millones de euros, y aproximadamente del 70 al 80 por ciento del valor se agrega

en la construcción naval. Además de ello, el presupuesto de defensa está aumentando y la construcción naval militar alemana se está convirtiendo en la industria de impulso para las innovaciones en la construcción naval en la era Shipbuilding 4.0.

En el Reino Unido tanto el UK Marine Industries Alliance, como The Society of Maritime Industries (SMI) trabajan en el apoyo y colaboración con el sector para la implantación de la Industria 4.0 en el mundo naval. Además, el Ministerio de Defensa ha elaborado una estrategia para definir las necesidades de la construcción naval militar en UK. (United Kingdom. Ministry of Defence, 2017). En esta estrategia se destaca que la implantación de las tecnologías digitales será clave de cara a conseguir un ahorro de costes y mejora de competitividad de la construcción naval militar británica.

En Holanda, el Instituto Holandés de Transporte y Construcción Naval (NISS) elaboró en el año 2016 el informe *Blueprint 2050: el mundo marítimo más allá del horizonte*, (NISS, 2016) en el que se presenta una visión de futuro de la industria marítima en 2050. En ellos se abarcan temas que están en la mesa de casi todos los grandes astilleros, como la automatización de tareas, el uso de nuevos materiales o fabricación aditiva, pero también se tratan otros muy ambiciosos, como por ejemplo el diseño automático de buques o los buques autónomos. Además, el astillero holandés, Damen, está inmerso en el desarrollo de numerosos proyectos relacionados con tecnologías 4.0. (DAMEN, 2018). Entre los proyectos en los que trabajan destacan los relacionados con el buque autónomo. Desde el año 2017 han estado trabajando (junto a otras 16 empresas) en el proyecto “Joint Industry Project (JIP) Autonomous Shipping”, que tiene como objetivo desarrollar la tecnología necesaria para que los buques puedan realizar ciertas operaciones de forma autónoma.

Las iniciativas digitales que se van desarrollando se integran en una plataforma única, a la que han nombrado “Damen Digital”, de esta forma los diferentes desarrollos pueden combinarse, dotando a la empresa de un enfoque más coherente en lo referente a digitalización. Además, esta plataforma puede servir como puerta de enlace, de forma que los datos que se recopilen de sus productos se procesen a través de esta plataforma, y sean accesibles también para los clientes, a través de servicios de IoT. La industria naval y marítima finlandesa también han elaborado, a través del grupo de trabajo creado para Construcción naval, en el año 2018, un informe del estado actual y futuro del

sector naval y marítimo en el país. (WP6-OECD, 2018) en el que una vez más se señala la necesidad de realizar más inversiones en digitalización, automatización y robotización de cara a fortalecer su competitividad. En el se destaca la voluntad del gobierno para apoyar a todos los sectores interesados.

### 1.2.1.2 Resto del mundo

Estados Unidos siempre se ha caracterizado por ser uno de los países constructores de los buques más tecnológicamente avanzados del mundo y en la carrera hacia la industria 4.0, la construcción naval no se puede quedar atrás. En el año 2016, Estados Unidos incrementó el objetivo de construcción naval militar, pasando de 308 a 355 buques al año, un objetivo muy ambicioso ya que se pretende que cada buque sea más complejo y tecnológicamente más avanzado que el anterior. (O'Rourke, 2018). Frente a objetivos como este, los astilleros se han planteado importantes cambios productivos, orientados a la creación de astilleros verdaderamente digitales.

Destaca especialmente el astillero Newport News Shipyard (NNS) que desde el año 2017 se encuentra inmerso en su proceso de transformación digital. (Dorwart, 2017). La compañía ha implantado una estrategia "sin planos en papel" diseñada para impulsar el trabajo hacia un astillero inteligente. La idea es eliminar todos los dibujos en papel o 2D del proceso de diseño y reemplazarlos con archivos digitales 3D. Con ello se conseguirá digitalizar un proceso manual del siglo XX para crear un sistema nuevo y optimizado que ayude a los operadores a trabajar de manera más inteligente y estar más sincronizados durante toda la construcción del buque. En su camino hacia la transformación digital, el astillero Newport New Shipbuilding también trabaja en líneas relacionadas con Realidad Aumentada, Modelado y Simulación y Escaneado Láser.

La construcción naval es uno de los 10 sectores industriales clave incluidos en la estrategia China: "Made in China 2025", junto a: Robótica, vehículos de próxima generación, energía, transporte ferroviario, aeronáutica y aeroespacial, nuevos materiales, medicina y agricultura. En el listado de los primeros proyectos piloto enmarcados en "Made in China 2025" aparece "Shipbuilding Intelligent Pilot Demonstration". (UK Trade & Investment, 2017).

En Corea, en el año 2015 se echó a andar un centro de innovación para el Sector Naval, en el que están representados los principales astilleros del país: Samsung Heavy Industries, Daewoo Shipbuilding and Marine Engineering y Hyundai Heavy Industries. Un ejemplo de los desarrollos que se están llevando a cabo en los astilleros coreanos se puede ver en el “buque conectado” de Hyundai Heavy Industries: Se recogen datos a bordo del buque en tiempo real, los cuales se transfieren a tierra, lo que permite a la compañía ofrecer servicios al cliente a lo largo del ciclo de vida del buque. (Spragg et al., 2017)

En Australia, existen importantes iniciativas, varias de ellas lideradas por el astillero BAE Systems Australia. Una de las líneas que se están tomando desde BAE Systems está relacionada con la construcción de un astillero digital en Adelaide. Buscan tener un nuevo astillero automatizado y digitalizado, que les permita llevar a cabo un diseño y construcción de buques más eficiente. La idea que se plantea es la de hilo digital: sincronización en toda la información de un proyecto, de forma que siempre exista una “única versión de la verdad” y que esté siempre actualizada, desde la fase de diseño, construcción y a lo largo de todo el ciclo de vida del buque (*Inside BAE Systems ' Digital Naval Shipyard*, n.d.).

Este mismo astillero colabora con el Innovative Manufacturing Cooperative Research Centre (Centro de investigación cooperativo de fabricación innovadora) y la Universidad de Australia del Sur para la aplicación de conceptos relacionados con el Astillero 4.0, como son: analítica de datos y big data, machine learning y nuevas técnicas de visualización. Para ello, además, realiza periódicamente convocatorias de innovación basadas en desafíos para fomentar la innovación tecnológica, como es el caso del proyecto “Digital Shipyard Track and Trace in the age of Industry 4.0”. BAE Systems Australia ha enmarcado estos desarrollos relacionados con el Astillero 4.0 en el entorno del programa de las Fragatas de la clase Hunter, con lo que buscan llevar a cabo una planificación y gestión más eficiente de este programa, así como disponer de un apoyo a la toma de decisiones (Ferguson, 2020).

Además de lo anterior, el gobierno regional de Victoria ha puesto en marcha una iniciativa denominada “Global Discovery Exchange” que busca ayudar a los constructores navales a avanzar hacia ese objetivo de digitalización en la industria naval, empleando como referencia lo que se está llevando a cabo en los astilleros estadounidenses, que se encuentran un paso por delante

de los australianos en lo que industria 4.0 aplicada a la construcción naval se refiere. Para ello se han creado acuerdos de colaboración entre astilleros australianos y estadounidenses.

### 1.2.1.3 España

En España, el Ministerio de Industria definió en el año 2019 la Agenda Sectorial de la Industria Naval, (Ministerio de Industria, 2019) en la que se destaca que mantener una industria naval puntera pasa por aprovechar las oportunidades que ofrecen las tecnologías digitales. Y siguiendo las recomendaciones de la UE, se sitúa la digitalización como el eje fundamental para mejorar la competitividad del sector.

Las principales iniciativas en torno al sector naval y marítimo español 4.0 están relacionadas con el astillero Navantia (Recamán Rivas, 2018). Desde esta empresa se han desplegado diferentes proyectos, apoyados en las diferentes tecnologías habilitadoras:

- Centro de Innovación en Tecnologías de Fabricación Avanzada, en colaboración con la Agencia IDEA, la Universidad de Cádiz, la empresa Airbus y el Clúster Marítimo Naval de Cádiz.
- Proyecto F4ST - Desarrollo de Tecnologías 4.0 para el Astillero del futuro El objetivo de este proyecto es la mejora de la competitividad de la empresa a través de 3 ejes: el Astillero 4.0, la transformación digital de los procesos actuales y la modernización las instalaciones.
- En la planta de Cartagena se llevan a cabo diferentes iniciativas 4.0 en colaboración con las Universidades Politécnicas de Cartagena y de Madrid.
- Unidad Mixta de Investigación (UMI) “Astillero del Futuro” formada por Navantia y la Universidade da Coruña.

A nivel autonómico, en Galicia, la Xunta a través de la Axencia Galega de Innovación (GAIN), junto con el Clúster del Naval Gallego (ACLUNAGA) y el centro tecnológico AIMEN, pusieron en marcha el proyecto “Shipbuilding 4.0” con el objetivo de dinamizar la capacidad innovadora del sector naval gallego, mediante el desarrollo de tecnologías de fabricación avanzadas que faciliten la tecnificación y evolución tecnológica del sector (AIMEN et al., 2020).

### 1.3 TECNOLOGÍAS 4.0

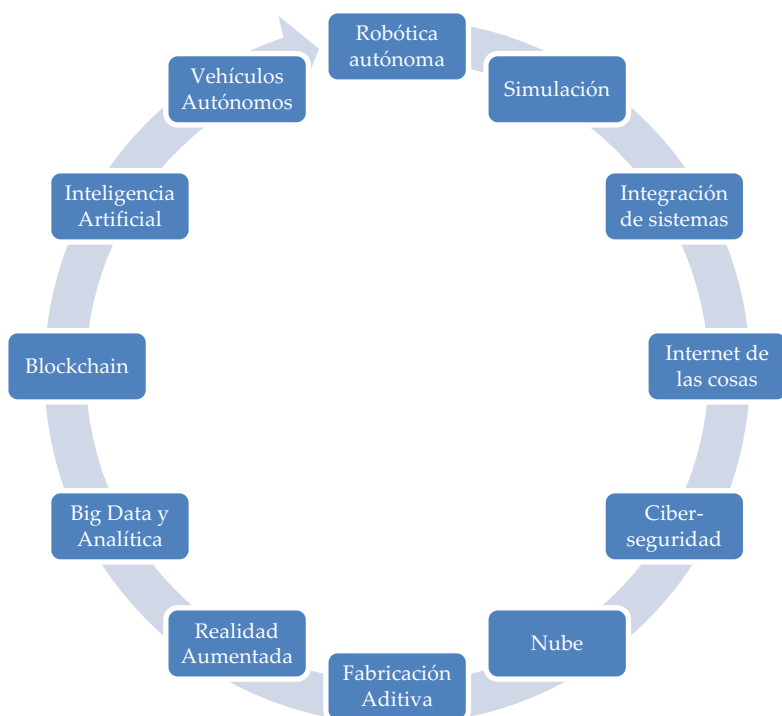
El Informe Boston Consulting habla de 9 tecnologías que están cambiando la producción industrial, y que son los pilares de la industria 4.0 (M. Rüssmann et al., 2019):

- Robótica autónoma.
- Simulación.
- Integración de sistemas (vertical y horizontal).
- Internet de las cosas Industrial.
- Ciberseguridad.
- La Nube.
- Fabricación Aditiva.
- Realidad Aumentada.
- Big Data y Analítica de datos.

Numerosa bibliografía hace referencia a estas 9 tecnologías habilitadoras como el camino a la hora de implantar la industria 4.0, aunque también se pueden encontrar autores que a estas 9 tecnologías añaden otras 3, (Gkerekos et al., 2019; Isaja & Calà, 2020; Lee et al., 2018; Sell et al., 2019):

- Blockchain.
- Inteligencia Artificial.
- Vehículos autónomos.

Agrupando lo anterior, en este trabajo se hará referencia a las siguientes 12 Tecnologías Habilitadoras, o KETs, que se agrupan en la imagen 2-1:



**Imagen 2-1: Tecnologías Habilitadoras (KETs)**

A continuación, se definirán brevemente estas tecnologías. En cada una de ellas se ha incluido una o más referencias representativas, en las que se analiza la relación entre esa tecnología habilitadora y el desarrollo de la Industria 4.0.

**Los robots** forman parte desde hace mucho tiempo de la estructura de funcionamiento de muchas plantas de producción. Se ha tratado tradicionalmente de robots programados para la realización de tareas repetitivas y en general dispuestos en zonas determinadas con estrictas medidas de seguridad. Si bien en los últimos años, debido al avance en la automatización de los robots el hecho de que estos sean cada vez más “conscientes de su entorno” debido a sensores, cámaras de visión, se ha llegado a desarrollar robots que interactúan entre ellos e incluso que lo hacen con seres humanos (cobots) (Gao et al., 2020).

Esto permite que se puedan estar introduciendo el uso de robots en un mayor número de tareas, ser más versátiles y servir de apoyo al operario en tareas que no son robotizables.



**La simulación**, entendiendo como simulación algo tan amplio que abarque desde simulaciones funcionales de elementos de un sistema, hasta simulaciones complejas de funcionamiento de una planta productiva, pasando por simulaciones 3D, de elementos finitos o CFD, son tecnologías que se llevan empleando años en las fases de diseño e ingeniería de productos y de procesos.

Si bien, en la actualidad se está dando un paso más, empleando las simulaciones a lo largo del ciclo de vida de un determinado producto, medio de producción o proceso industrial (Rodič, 2017). Poder simular en tiempo real unido al hecho de poder analizar y comparar los resultados obtenidos en las simulaciones con lo que se está obteniendo en la realidad permitirá obtener múltiples beneficios:

- Identificar problemas de funcionamiento.
- Adelantarse a los fallos que se puedan producir evitando o reduciendo costes de mantenimientos y reparaciones.
- Proponer mejoras de diseño de un producto o proceso.

Hasta hoy, la mayoría de los sistemas relacionados con las tecnologías de la información (IT) no están completamente integrados. Pocas veces las industrias, sus suministradores y sus clientes, están estrechamente conectados. Y lo mismo sucede a menudo con los departamentos de ingeniería, producción y servicio dentro de la misma empresa. En el entorno de la Industria 4.0 se busca que las empresas, departamentos, y funciones se vuelvan mucho más cohesivas, una integración transversal de datos que permitan cadenas de valor automatizadas (Chukalov, 2017).

Si una tecnología de la Industria 4.0 destaca sobre las demás, esta es, sin duda, el **Internet de las Cosas, o IoT** (Malik et al., 2021). La posibilidad de contar con interconexión de múltiples dispositivos, tanto en una planta de producción como en el funcionamiento de un producto es el punto de partida para que el resto de las tecnologías 4.0 puedan aplicarse.

El IoT ha permitido la conexión en tiempo real de máquinas y productos, ya sea para transferencia de información entre ellos (máquina a máquina) o para transferencia de datos a personas. A menudo, en el entorno industrial se habla de IIoT (Industrial Internet of Things) (Diaz-Casas et al., 2018; Sisinni et al., 2018) en lugar de IoT.

En una industria cada vez más conectada, con miles de dispositivos emitiendo y recibiendo datos, será de vital importancia blindar la seguridad de toda esta información digital (**ciberseguridad**). (Rubio et al., 2018; Waslo et al., 2017).

La tecnología a la que se ha denominado **Nube** es aquella que permite ofrecer servicios de computación totalmente online, es decir que no requieran de una instalación local. Puede tratarse de simples servicios de almacenamiento de datos, o de complejos programas de diseño, control, gestión, etc.(Red Repply, 2020).

**La fabricación aditiva** mediante impresoras 3D permite fabricar piezas complejas en un tiempo razonable y a un coste competitivo. Además, permite eliminar los tramites logísticos, que pueden presentarse en numerosas ocasiones (no disponibilidad de una pieza, tiempos de entrega elevados, etc.) (Dilberoglu et al., 2017).

**La realidad aumentada** permite mostrar información en tiempo real superpuesta a la realidad física, que puede servir de soporte al proceso productivo. Además de ello, la realidad aumentada puede ayudar en la realización de tareas de mantenimiento de máquinas o equipos, visualizando sobre la realidad las diferentes secuencias de trabajos a realizar. Permitiendo, además, que el encargado de la realización de las tareas no tenga que ser necesariamente un experto (Masood & Egger, 2019).

**Big data y analítica:** Las tecnologías de análisis de datos y de análisis de datos masivos (big data), permitirán recopilar y analizar en tiempo real un elevado número de señales ya sea de una fábrica, de una máquina o de funcionamiento de un producto. Con el análisis inteligente de diferentes señales, será posible adelantarse a la toma de decisiones, predecir fallos en una máquina o incluso mejorar la productividad de la planta (Greco et al., 2019; Singh, 2020).

**El Blockchain** (cadena de bloques) nació en el año 2018, como la arquitectura que permitió la aparición del Bitcoin. Se trata de una base de datos distribuida y cifrada. Un gran libro de cuentas que solo puede ser actualizado con el consenso de la mayoría de los participantes y en el que, además, una vez introducida, la información nunca puede ser borrada.

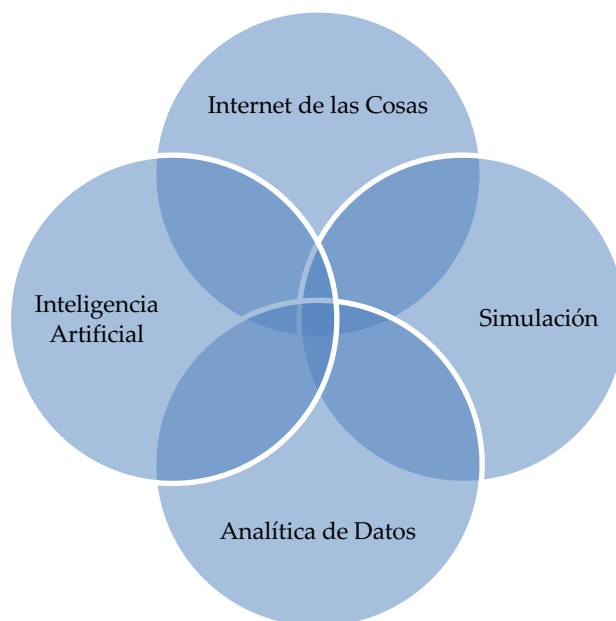
En la industria, su aplicación puede ser de enorme utilidad en la trazabilidad en la cadena de suministro (Lim et al., 2021), y también en IoT (Q. Wang et al., 2020). La gestión de datos industriales, separados en cadenas de bloques privadas y públicas, permitirá ganar eficiencia y seguridad en su tratamiento.

Ligado al avance del IoT, la implementación de técnicas basadas **en Inteligencia artificial** permitirá lograr importantes mejoras productivas y operativas. El uso de técnicas como el machine learning, podrá aplicarse, por ejemplo, a mantenimiento predictivo, soporte a la toma de decisiones, optimización de procesos, etc. (Lee et al., 2019).

La última tecnología que se considera habilitadora de la Industria 4.0 se basa en el uso de **vehículos autónomos**. Y dentro de vehículos autónomos de aplicación en la industria, se pueden encontrar vehículos de guiado automático, AGVs, de aplicación a múltiples industrias (de hecho, en muchas ya están ampliamente implementados) drones (sobre todo para tareas de inspección de trabajos, o para su uso en seguridad), e incluso vehículos submarinos (los cuales se pueden emplear, por ejemplo, en el sector offshore) (Sell et al., 2019).

#### **1.4 TECNOLOGÍAS 4.0 PARA GEMELO DIGITAL**

El despegue de la Industria 4.0 y en especial los desarrollos relacionados con el Gemelo Digital no habrían sido posibles sin el desarrollo de las tecnologías de IoT, que permiten tener sistemas conectados en tiempo real, sin la simulación de sistemas y de procesos y sin la analítica de datos y la inteligencia artificial. La combinación correcta de estas tecnologías permitirá el despliegue de un Gemelo Digital completo.



**Imagen 2-2: Tecnologías clave en el desarrollo del gemelo digital**

### **1.4.1 Internet de las Cosas en la Industria.**

El Internet de las Cosas, IoT, es el componente tecnológico fundamental sobre el que sienta sus bases el paradigma de la Industria 4.0 (Malik et al., 2021; Vega et al., 2015). El término Internet of Things (Internet de las Cosas, IoT) fue propuesto por Kevin Ashton en 1999 (Madakam et al., 2015; Suresh et al., 2014) mientras preparaba una presentación en su empresa (Procter & Gamble) sobre un desarrollo que había hecho que se basaba en el uso de etiquetas RFID (radio-frecuencia) en y otro tipo de sensores en productos de la cadena de suministro, de modo que las etiquetas y sensores podrían generar información sobre dónde están los productos (si se habían escaneado a la entrada de un almacén, o si habían sido colocados en una estantería o vendidos). Lo realmente novedoso de esta cuestión es que hasta el momento la información disponible en internet se había producido casi exclusivamente por individuos o por sistemas, pero no por cosas físicas.

Este desarrollo le permitió ir al MIT a continuar su trabajo fundando un consorcio de investigación de RFID llamado Auto-ID Center. Este centro se creó con el objetivo de crear un sistema global de estándares abiertos para

implantar sistemas de identificación y sensores por radiofrecuencia. Pero en lo que se refiere al término “Internet of Things” no fue hasta la segunda mitad de los 2000’s cuando se empezó a generalizar, debido a que el uso de elementos conectados a internet empezó a ser una realidad a nivel global.

Se puede definir el internet de las cosas como aquel entorno tecnológico en el que los objetos cotidianos están conectados a internet y además de ello son capaces de recibir, generar y enviar información.

Se estima que en el año 2023 habrá cerca de 30 mil millones de dispositivos electrónicos conectados a Internet en el mundo (CISCO, 2020). En unos cuantos años, internet de las cosas será una estructura compleja, en donde se espera que haya miles de millones de nodos capaces de medir prácticamente cualquier cosa, así como de extraer y transferir información.

El origen de los dispositivos conectados se puede situar en el S XIX, cuando se llevaron a cabo los primeros experimentos de telemetría. Uno de los primeros, realizado por científicos franceses en el año 1874 se basó en instalar dispositivos de medición meteorológica y de espesor de nieve en Mont Blanc, los datos recogidos se transmitieron a París a través de un enlace de radio corta. Después se fueron completando estudios similares en Rusia o EEUU, avanzando de forma notable en el uso de la telemetría y, en general, en el uso de las tecnologías de la comunicación. (Hewlett Packard, 2019).

El uso del IoT en la industria, permite disponer de máquinas y de productos conectados, incluso de fábricas completas. Las principales ventajas que se presentan a la hora de implantar tecnologías IoT en la industria están relacionadas con el aumento de la velocidad de la transferencia de información, la monitorización y optimización de procesos, de cara a la mejora de productividad (reducción de tiempos inoperativos, ya sea por planificación o por paradas no planificadas) así como para optimización de la gestión energética.

Entre las principales ventajas del uso del IoT industrial, teniendo en cuenta que este puede estar aplicado a un producto o a un proceso, se debe destacar:

- Ahorro de costes y aceleración de la producción.
- Mejora de la productividad, debido a la reducción de tiempos de fabricación, en el caso de los procesos y de ahorro de tiempos inoperativos en el caso de un producto.
- Mejora de la trazabilidad y de la localización de activos.

- Optimización de procesos, ya sea de los procesos de fabricación o del proceso operativo de un producto.
- Valor añadido que aporta la fabricación inteligente, o la gestión inteligente de un producto.
- Uso proactivo de los datos.

### 1.4.2 Simulación

En el entorno de la Industria 4.0, la simulación se presenta como una potente herramienta que permite ensayar y conocer el funcionamiento de un sistema, antes incluso de que se haya fabricado (Rodič, 2017; Wortmann et al., 2017). La simulación se basa en aplicar programas informáticos que permiten replicar el comportamiento de un sistema, o un conjunto de sistemas, de forma que se puede comprobar cómo se comporta.

Robert Shannon definió en los años 70 la simulación como (Shannon & Johannes, 1976): "el proceso de diseñar un modelo de un sistema real y llevar a término experiencias con él, con la finalidad de comprender el comportamiento del sistema o evaluar nuevas estrategias -dentro de los límites impuestos por un cierto criterio o un conjunto de ellos - para el funcionamiento del sistema".

El mundo de la simulación es enorme, ya que las simulaciones pueden tener el grado de detalle, la naturaleza y el objetivo que se desee. Se puede simular un fenómeno físico, el funcionamiento de una máquina, de un proceso de fabricación, etc. Se puede hablar de simulación cuando se hace referencia simplemente a un pequeño programa en el que se representa el movimiento de un péndulo al que se le aplica una fuerza. Pero también se habla de simulación cuando el modelo considerado es un sofisticado software que permite emular el pilotaje que un vehículo aeroespacial.

Las principales ventajas que aporta la simulación se pueden resumir en lo siguiente. La simulación:

- Permite adquirir experiencia y conocimiento sobre el funcionamiento del sistema que se está simulando.
- Permite identificar problemas en un proceso complejo, en que empleando únicamente la experiencia o el conocimiento experto no es suficiente.

- Permite plantear escenarios alternativos en un sistema.
- Permite conocer el comportamiento de sistemas complejos, para los cuales fabricar un prototipo no sería abordable ni económica ni tecnológicamente.
- Se puede aplicar en cualquier fase del ciclo de vida del sistema que esté considerando, desde el diseño, construcción y hasta la operación de este.

### 1.4.3 Inteligencia Artificial

El término “inteligencia artificial” fue acuñado por primera vez, por el informático estadounidense, John McCarthy, en el año 1956 durante la Conferencia de Dartmouth que él mismo organizó y donde se puede considerar que nació esta disciplina. (J. McCarthy et al., 1955). Se puede definir la inteligencia artificial como aquella disciplina, dentro del campo de la informática, que busca la creación de máquinas que puedan imitar comportamientos inteligentes.

Se dice que una máquina o un sistema es inteligente si es capaz de percibir su entorno (sensores, visión, etc.) y que además lleva a cabo acciones que permitan (o al menos que aumenten las probabilidades) de llevar a cabo una tarea o lograr un objetivo. El término inteligencia artificial se utiliza cuando una máquina imita funciones cognitivas propias de seres humanos, como pueden ser: percibir, razonar, resolver problemas o aprender.

Dentro de la inteligencia artificial existen diferentes campos de estudio: robótica, sistemas expertos, reconocimiento del habla, procesamiento del lenguaje natural, reconocimiento visual, reconocimiento de texto, machine learning y deep learning, etc. A la combinación de estas tecnologías se le denomina Inteligencia Cognitiva. Se considera que la unión de reconocimiento visual, sonoro, comprensión lectora, NLP y machine learning permitirá crear sistemas que puedan comprender e interactuar de forma similar a los humanos.

En el entorno de la industria 4.0, y en concreto, en el desarrollo del gemelo digital, la rama de la inteligencia artificial que más relevancia tiene es, sin duda, el aprendizaje automático o machine learning. Se trata de la rama, dentro de la inteligencia artificial, que busca cómo dotar a las máquinas de capacidad de aprendizaje, es decir, que sean capaces de generar conocimiento a partir de experiencias, en contraposición a las técnicas tradicionales de IA en las que el

conocimiento se “programa”. El machine learning nació en los años 50, con el objetivo de estudiar el reconocimiento de patrones y aprendizaje de máquinas.

Arthur Samuel, uno de los pioneros en la IA, se propuso “enseñar” a una máquina a jugar a las damas, y escribió el que se puede considerar como el primer programa de aprendizaje informático. No se trataba de que la máquina calculase todos los movimientos posibles cada vez que era su turno (como se podría hacer con la implementación de un sistema experto), sino que se adaptara a las circunstancias concretas de cada movimiento de su rival. Las instrucciones que se le daban eran únicamente unas cuantas pautas, mientras que la clave estaba en que el programa era capaz de aprender de sus propios errores. En los años 60 este programa ya era capaz de derrotar a los campeones del juego (Samuel, 1960).

El funcionamiento del machine learning se basa en el reconocimiento de patrones complejos compuestos por un elevado número de parámetros. Es decir, un algoritmo de programación que es capaz de modificarse con los datos que recibe de entrada, es decir, contará con autonomía suficiente para que aprenda de sus errores u optimice sus aciertos, sin necesidad de que en este proceso intervenga un humano. Se pueden distinguir 3 tipos de aprendizaje:

- Aprendizaje supervisado, es aquel en el que al algoritmo se le suministran los datos previamente etiquetados, es decir, las entradas y las salidas correspondientes. A partir del conocimiento de estos problemas resueltos, el algoritmo será capaz de resolver nuevos problemas. Un ejemplo de aprendizaje supervisado serían las redes neuronales de tipo backpropagation.
- Aprendizaje no supervisado, el algoritmo únicamente recibe los patrones de entrada y unas indicaciones previas. A partir de ahí extrae características que le permiten agrupar esos patrones, creando diferentes clases en función de su clasificación.
- Aprendizaje por refuerzo, en este caso la máquina aprende basándose en el éxito o fracaso que obtenga. Los algoritmos en este caso cuentan con información previa, pero se dedican a aprender por su cuenta en base a la tasa de éxito que vayan obteniendo.



#### 1.4.4 Analítica de Datos y Big Data

El IoT ha permitido el acceso en tiempo real (o casi) a una cantidad de datos impensable hace pocos años. Esta generación de datos es cada vez más grande. Pero los datos, por si solos no sirven de demasiado. Es necesaria una gestión de esos datos que aporte valor. Es lo que se entiende como analítica de datos, o Big Data, cuando los volúmenes de datos a considerar son extraordinariamente grandes.

El primer paso a la hora de definir un proyecto de analítica de datos es definir y focalizar el objetivo, por ejemplo: reducir en un porcentaje determinado las paradas para mantenimiento de una planta o de un sistema, aumentar la producción de una planta o mejorar la calidad de un producto que se está fabricando. Una vez definido el objetivo, los pasos que se suelen citar a la hora de definir la secuencia del análisis son (Belhadi et al., 2019):

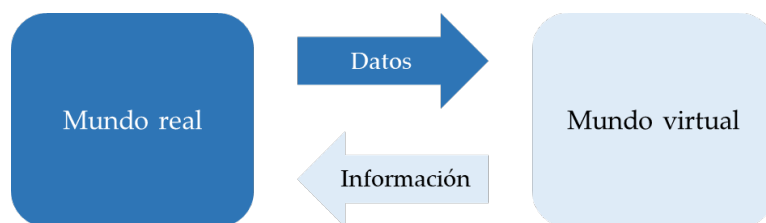
- Etapa de descripción: En esta etapa se informa acerca del proceso: ya sea un proceso productivo, o de funcionamiento de un sistema. Se genera información descriptiva y/o estadística sobre: producción, paradas, mantenimientos, fallos, etc.
- Etapa de diagnóstico: En esta fase lo que se pretende es determinar por qué sucede aquello que se ha definido en la etapa de descripción. Se realiza por lo tanto un informe en el que se recogen aquellos eventos identificados con los datos que provocan ese evento.
- Etapa de predicción: una vez recopilada la información de las 2 etapas anteriores, se estará en posición de llevar a cabo una predicción de eventos a partir de la observación de los datos que se están obteniendo.
- Etapa de prescripción: en esta etapa final y a partir de todo lo anterior, se le podrá indicar a la máquina como debe reaccionar ante determinados eventos e imprevistos.

La gestión inteligente de los datos que genera el gemelo digital podrá emplearse principalmente para lo siguiente:

- Establecer patrones de uso o de funcionamiento de un sistema, equipo, etc. que permita optimizar su uso: desde un punto de vista de gestión energética, de productividad, etc.
- Evaluar el estado de salud/deterioro de los componentes del sistema.
- Realizar modelos de predicción de fallos (mantenimiento predictivo).

## 1.5 GEMELO DIGITAL

El concepto de gemelo digital nació en los años 70 en la NASA, cuando la agencia espacial empezó a llevar a cabo simulaciones del comportamiento de las naves o equipamientos, con el objetivo de garantizar la viabilidad de las misiones y velar por la seguridad de las tripulaciones. Aunque no es hasta 2002 cuando se habla de este concepto de gemelo digital, en una conferencia en la Universidad de Michigan sobre Gestión de Ciclo de Vida del Producto, o PLM (Product Lifecycle Management), en la que Michael Grieves, se refirió a algo que en ese momento denominó como “Conceptual Ideal for PLM” pero que contenía los 3 elementos principales que se considera que definen lo que hoy se conoce como gemelo digital, y que son: El activo físico en el mundo real, el activo virtual en el mundo virtual y los flujos de datos e información entre ambos mundos (Michael Grieves & Vickers, 2016).



**Imagen 2-3: Representación de Michael Grieves sobre el gemelo digital y el flujo de datos e información entre el activo físico y el virtual**

Grieves planteaba que los futuros problemas operativos comenzaban, en realidad, en las fases de diseño y producción, y que se incrementaban con la interacción humana. Para mitigar esto propuso el concepto de gemelo digital que conectase el sistema físico con su virtual equivalente. De la colaboración con Grieves parte el planteamiento de la NASA de utilizar el concepto de gemelo digital (Glaessgen & Stargel, 2012) como un nuevo paradigma a aplicar a sus futuros transbordadores. El reto al que se enfrentaban era que los nuevos vehículos debían ser más ligeros, pero, al mismo tiempo, se enfrentarían a condiciones más extremas de funcionamiento en periodos más largos. Las soluciones hasta ese momento se basaban en distribuciones estadísticas o pruebas físicas que permitiesen asumir que había similitud entre lo que se testeaba y la operativa real. Ante la limitada fiabilidad de esta aproximación, propusieron un cambio de paradigma, el gemelo digital, de forma que integrase una

simulación fiable de los vehículos a través de un sistema de gestión de la integridad de este, históricos de mantenimiento y de toda la información disponible de la flota. Esto permitiría crear un espejo del vehículo real en el mundo digital y aumentar los niveles de seguridad y fiabilidad.

### 1.5.1 El concepto de gemelo digital

La definición inicial de gemelo digital que propusieron Grieves y Vickers es: "El gemelo digital es un conjunto de construcciones virtuales de información que describe de forma completa un producto físico existente o potencial desde el nivel microatómico hasta el nivel macrogeométrico. De forma óptima, cualquier información que pueda obtenerse a partir de la inspección de un producto fabricado físicamente, puede obtenerse desde su gemelo digital" (Michael Grieves & Vickers, 2016).

Cabos y Rostock definen como las características principales que debe de tener un gemelo digital (Christian Cabos & Rostock, 2018):

- La primera propiedad está directamente relacionada con el término "gemelo": El concepto de gemelo sugiere que debe existir un segundo objeto, físico, que es "reflejado" por el gemelo digital.
- La segunda propiedad es que el gemelo digital debe reflejar cambios en el gemelo físico. De lo contrario, perdería su propiedad "gemela".
- La tercera propiedad es que la existencia de un gemelo digital ha de tener un propósito que genere un efecto positivo sobre el objeto físico. Por lo tanto, debe tener alguna lógica inherente o conectada, como por ejemplo la capacidad de simulación, para cumplir con ese propósito.

El grado de detalle de un gemelo digital dependerá del propósito para el que se haya pensado: del contexto para el que está hecho el gemelo digital y en el que se pretende utilizar. Sobre la base de las propiedades anteriores, se propone que tres elementos básicos sean los constituyentes de un gemelo digital:

- Una representación digital del objeto, que se relaciona con la estructura del sistema. Tales representaciones podrían tener una interfaz de usuario visual, por ejemplo, en forma de modelos 3D, o pueden consistir en especificaciones del sistema que describan la arquitectura del sistema, incluidos los componentes y sus interfaces.

- Modelos de comportamiento, que describen la respuesta dinámica del sistema (es decir, su estado y sus resultados). Dichos modelos pueden incluir la operación del sistema (para simular cómo el sistema realiza las funciones para las que se ha diseñado), la respuesta estructural a las cargas a las que está expuesto, así como los procesos de degradación y daño.
- La configuración o condición del sistema: son registros de datos del estado pasado y actual del sistema, o cambios en él, capturados a través de sensores, inspecciones, etc.

## **1.6 GEMELO DIGITAL EN LA CONSTRUCCIÓN NAVAL E INDUSTRIA MARÍTIMA**

Un gemelo digital es una representación digital de un objeto físico. Para cada barco físico (incluidos los barcos gemelos individuales), debe haber una, y solo una, réplica digital exacta. Si la nave física cambia, el modelo digital también debería actualizarse.

Tal y como detalla D. Morais, CEO de SSI, los datos y atributos geométricos del buque que se originan en la herramienta CAD de ingeniería se incorporan en su gemelo digital, pero el concepto completo abarca mucho más. El gemelo digital contendrá detalles sobre los requisitos del barco vinculados a piezas y ensamblajes, además de vínculos a elementos como simulaciones y cálculos e incluso información sobre, por ejemplo, pintura. También incluirá la información proporcionada por los proveedores correspondientes a todos los componentes, es decir, a cada motor, válvula y bomba, etc. Incluso los detalles de los procesos de producción específicos utilizados para la secuenciación de soldadura y ensamblaje deben estar en el gemelo junto con información sobre los centros de trabajo. En otras palabras, si hay datos sobre cómo está construido el buque, debería incluirse como parte de un gemelo digital completo (Morais, n.d.).

Pero no se detiene ahí. Después de la entrega de un barco, la información debe seguir acumulándose. El gemelo digital deberá incluir registros de servicio, reparaciones y modernizaciones, documentación de eliminación de incrustaciones. Además, deberá incluir datos reales de los sensores que reflejen lo que está sucediendo en todo el barco.

En resumen, el gemelo digital debe incluir todos los elementos del buque tales como sus sistemas, equipos, la estructura, etc., agrupando la información disponible sobre el barco en un mundo digital. De esta forma puede contener modelos digitales e información relacionada con el barco (Smogeli, 2017), como, por ejemplo:

- Modelos analíticos hidrodinámicos y para estructuras.
- Modelos de simulación.
- Modelos de información para sistemas y componentes.
- Modelos de visualización 3D tanto de estructuras como de componentes.
- Modelos en el dominio del tiempo para componentes y sistemas.
- Datos reales recopilados de sensores y datos de estado del barco en tiempo real.
- Algoritmos de control.
- Redes de comunicación.
- Información digital: documentación e informes.
- Documentación y procedimientos relacionados con la certificación o con la clasificación del buque.

La idea de gemelo digital del buque ha ido evolucionando desde los años 90 en los que surgieron los primeros conceptos básicos. En la siguiente tabla se puede ver de forma resumida como ha sido la evolución en el tiempo de los gemelos digitales, considerando que se pueden hablar de 6 generaciones (Erikstad, 2018), que se resumen en la siguiente tabla:

**Tabla 2-1: Evolución temporal de las diferentes tipologías de gemelos digitales.**  
Adaptada de (Erikstad, 2018).

PERÍODO	TIPO	CARACTERÍSTICAS
1990 - 2000	Modelos de producto	Una sola fuente de información estructurada del elemento a reproducir.
2000 - 2015	Captura de datos mediante sensores	Big data.
2015 - 2018	Gemelos digitales	Modelos basados en datos (data-driven) en tiempo quasi-real.
2018 - 2020	Gemelos digitales con capacidad de aprendizaje	Modelos basados en datos en tiempo real, que incluyen técnicas de inteligencia artificial y/o machine learning para reproducir el sistema físico.
2020 - 2022	Gemelos digitales predictivos	Modelos basados en datos en tiempo real, que reproducen completamente el sistema físico.
2022- 2024	Gemelos digitales autónomos	Modelos basados en datos en tiempo real, con capacidad de predicción y control sobre el sistema físico.

Una de las primeras iniciativas surgió a mediados de los años 90. DNV desarrolló el proyecto “Ship Product Model” en el que se trataba de recopilar en un repositorio común toda la información de los buques de su flota, de forma que esta estuviese siempre actualizada y pudiese ser accesible de forma inmediata por cualquiera que tuviese acceso a ese repositorio. Se trataba, por lo tanto, de un modelo digital que reflejaba un activo real, pero sin capacidad de observación en el modelo real ni cálculo de comportamiento en tiempo real.

El avance que se produjo en el desarrollo de sensores, así como en tecnologías de comunicación propició que los sistemas del buque pudiesen cada vez más conectados. Los armadores comenzaron a recopilar petabytes de datos de cientos de sensores a bordo, aunque, la mayoría de las veces sin las herramientas, métodos y modelos necesarios para convertir esos datos en información útil para respaldar la toma de decisiones en el mundo real. Además, durante

los últimos años, la ciencia de datos y el aprendizaje automático se han convertido en tecnologías clave, ofreciendo una oportunidad para extraer valor.

Según K. Bruun, (Bruun, 2018), implantar gemelos digitales en buques y en general en la industria naval y marítima, permitirá:

- Una solución basada en la nube donde todos los datos, modelos e información relevantes del barco estarán disponibles en cualquier momento.
- Poseer información actualizada en tiempo real durante todo el ciclo de vida del barco, desde la fase de diseño hasta el desmantelamiento.
- Disponer de modelos de simulación actualizados y refinados con datos operativos reales y datos de sensores de modo que reflejen de forma fidedigna la realidad.
- Incorporación y almacenamiento de datos específicos de operaciones externas, como el clima, las corrientes y las olas.
- Una plataforma de colaboración que evita que las partes interesadas trabajen en silos independientes y no intercomunicados.
- Realizar el intercambio controlado de datos, modelos e información actualizada entre las partes interesadas.
- Tener un conocimiento temprano de posibles conflictos de diseño.
- Alcanzar una mejor calidad y eficiencia de los procesos de aprobación y certificación.

### **1.6.1 Algunas aplicaciones reales**

La aplicación del concepto de gemelo digital a la industria marítima es escasa. Aún así, existen algunos desarrollos, como es el caso de la Open Simulation Platform (OSP), (OSP, 2018; Perabo et al., 2020). La Rolls-Royce Marine, la Norwegian University of Science and Technology (NTNU), SINTEF Ocean y DNV GL colaboran en el desarrollo de una plataforma digital para su uso en buques. El principal objetivo que busca esta plataforma es establecer un estándar para modelos y simulaciones dentro de la industria marítima que permita la reutilización de modelos y las simulaciones de sistemas colaborativos. Con ello se busca poder resolver importantes desafíos en el diseño, construcción y operación de buques. Para ello, se basan en 3 casos de uso: el diseño de un sistema de propulsión de ferry híbrido, la puesta en servicio virtual de una

embarcación de servicio costero y la planificación de una operación de grúa submarina.

En 2016, DNV GL presentó el concepto NauticusTwinity, una plataforma de simulación para el desarrollo de gemelos digitales para el diseño, prueba y verificación a lo largo del ciclo de vida de un buque (Bruun et al., 2016).

Otro ejemplo es el caso de la Autoridad Marítima Danesa, quienes buscan impulsar el concepto de gemelo digital dentro del clúster marítimo con el que cuentan y que se denomina “Blue Denmark” (Bruun, 2018).

El centro tecnológico marino y offshore de Singapur (TCOMS), junto con MPA Singapore y Keppel Offshore & Marine, trabajan en el desarrollo de un gemelo digital de un remolcador con el objetivo de optimizar las operaciones del mismo a través de la analítica de datos y de herramientas de visualización. Para ello, será necesario desarrollar modelos para la representación de la parte física y usar simulaciones para control y predicción del comportamiento del buque (Keppel, 2019). Algo similar plantean investigadores de la Universidad de Michigan quienes pretenden emplear la tecnología de gemelo digital para poder predecir los movimientos del buque, y en esta caso, además, el daño que se pueda generar por fatiga (Schirmann et al., 2019).

Desde el punto de vista educativo se ha trabajado en el NTNU en el desarrollo de un gemelo digital de un buque de investigación oceánica (Asbjørnslett et al., 2018).

Con lo anterior, se demuestra que un gemelo digital irá más allá de una simple copia digital de un barco real, sino que será un entorno digital que agrupe toda la información relacionada con el buque y con su entorno y que, además, le aporte coherencia y estructura de tal forma que se obtenga un conjunto con capacidades superiores a las que se podrían llegar a alcanzar considerando la agregación de cada parte individual.

## **1.7 GEMELO DIGITAL EN LA CONSTRUCCIÓN NAVAL MILITAR**

La evolución de los buques de guerra está dando lugar a flotas cada vez más avanzadas y complejas tecnológicamente. Además de ello, las armadas



piden que esos buques duren más y que sean capaces de adaptarse a los cambios que se les exijan durante esa vida útil prolongada.

Asegurar la adaptabilidad necesaria a las demandas y los roles cambiantes que se les puede asignar en más de 50 años de servicio inevitablemente implicará actualizaciones y readaptaciones, que podrían ser donde el concepto de gemelo digital realmente comience a cobrar fuerza.

En este contexto, las principales armadas del mundo y los principales astilleros de construcción naval ya están contemplando en mayor o menor medida el desarrollo de gemelos digitales.

La armada de Estados Unidos está en una fase temprana en su desarrollo de gemelo digital, aunque las iniciativas a este respecto van encaminadas principalmente a la capacidad de simular buques antes de su construcción y al mantenimiento predictivo. La Oficina de Investigación Naval (ONR: Office of Naval Research) trabaja en lo que ha denominado como “Navy Platform Digital Twin”. Los trabajos desarrollados en esta plataforma se basan en la fusión de datos medidos y modelos basados en la física de buques de guerra. El objetivo de esta investigación es predecir el rendimiento óptimo, el estado del material y la susceptibilidad de un buque de guerra. El objetivo de esta plataforma es crear un marco basado en modelos que proporcione información que permita una mejor asignación de recursos, planificación logística y toma de decisiones in situ (Fu, 2017).

Por otro lado, el American Bureau of Shipping (ABS) está construyendo un Digital Twin para la flota MSC (U.S. Navy’s Military Sealift Command) de la Armada de los Estados Unidos. El objetivo de los gemelos digitales es detectar comportamientos anormales, actuando así como un sistema de alerta temprana de problemas (Lewis, 2018). Para ello se están recogiendo datos de los sensores instalados en el casco y maquinaria de los buques de esta clase, desarrollando un gemelo digital para cada uno de los buques. Con esta inteligencia para todo el barco, se obtendrá una visión holística de la salud estructural de todo el buque y el rendimiento en tiempo real de los equipos a bordo.

También en Estados Unidos, la entidad NAVMAR (Naval Information Warfare Systems Command) creó un grupo de trabajo propio para los temas relacionados con gemelo digital, siendo este tema uno de los 7 ejes principales en los que trabaja NAVMAR, con el objetivo de implementar una plataforma

de información en los buques de la armada estadounidense. (Chan & Kumar, 2019). De hecho, ya han completado el desarrollo de lo que han denominado como “modelo digital de sistemas” para implementar en el buque USS Abraham Lincoln, y que es un gemelo digital (Digital Lincoln) en el que se integran 5 sistemas, interconectados entre sí. (NAVWAR, 2019).

- Distributed Common Ground System-Navy (DCGS-N): un programa que proporciona la inteligencia primaria, la vigilancia, el reconocimiento y la capacidad de apoyo al objetivo. Son herramientas críticas para la conciencia del espacio de batalla.
- Navy Integrated Tactical Environmental System-Next Generation (NITES-Next): un sistema que utiliza datos de meteorología y oceanografía para ayudar con la planificación y la ejecución de la misión, así como con la toma de decisiones críticas y el conocimiento de la situación.
- Maritime Tactical Command and Control (MTC2): Un programa de mando y control que ofrece ayudas de gestión de batalla para planificar, dirigir, supervisar y evaluar de forma dinámica las operaciones.
- Global Command and Control System - Maritime (GCCS-M): un sistema que: fusiona, correlaciona, filtra, mantiene y muestra información de ubicación y atributos de fuerzas terrestres, marítimas y aéreas amigas, hostiles y neutrales.
- Agile Core Services (ACS): proporciona la infraestructura para soportar la migración de aplicaciones.

Existe una propuesta de cara al futuro, en la cual el equipo que ha desarrollado el Digital Lincoln aprovechará las lecciones aprendidas y las aplicará al desarrollo de un gemelo digital de sistema de sistemas del USS Dwight D. Eisenhower (CVN 69). A medida que la Armada continúa en su viaje de transformación digital, el objetivo es crear modelos digitales de todos los sistemas en todas las plataformas para mejorar la ciberseguridad, mejorar la capacidad del sistema, aumentar la velocidad de entrega de tecnología y reducir el tiempo y el costo de instalación.

Otro país en el que se está avanzando en la aplicación de gemelos digitales a buques militares es Australia. La Royal Australian Navy en colaboración con la empresa Kongsberg Digital está llevando a cabo el desarrollo de un

sofisticado simulador de puente, denominado K-Sim Navigation (Voith, 2018). Esta plataforma permite simular diferentes escenarios de forma realista, incluyendo un modelado hidrodinámico de última generación que permite que las embarcaciones, los objetos y los equipos se comporten e interactúen como en la vida real.

En Canadá, la Royal Canada Navy ha elaborado una estrategia para la digitalización de su flota, entre la que se encuentra como actividades clave el desarrollo de gemelos digitales (RCN, 2018).

En España, los principales desarrollos en cuanto al despegue del gemelo digital en la industria naval militar giran en torno al astillero Navantia, aunque la Armada Española también está involucrada en este proceso. En la Armada Española cabe destacar el trabajo realizado por el CESADAR (Centro de Supervisión de Datos de la Armada) en el que a día de hoy ya se recogen datos de alrededor de 35 buques de la flota española (con una media de 3000 señales por buque).

En Navantia, el desarrollo del gemelo digital gira en torno al programa de las fragatas F110, buscando con su implantación unas funcionalidades entre las que se pueden destacar las siguientes (Navantia, 2020):

- Representar virtualmente al buque en tiempo útil.
- Soportar servicios de información y gestión avanzados.
- Soportar asistentes inteligentes a la toma de decisión como para la eficiencia de operación o como para la seguridad interior.
- Soportar agentes inteligentes en determinados casos de uso de interés.



### 3. DESARROLLO METODOLÓGICO PARA IMPLANTACIÓN DEL GD

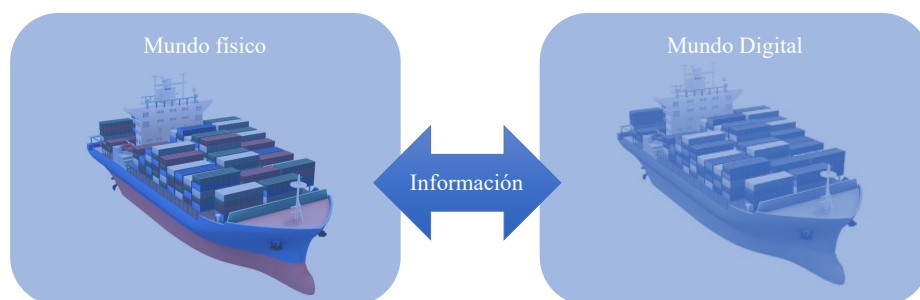
---

A lo largo de este capítulo se analizará la metodología a seguir para llevar a cabo la implementación de un gemelo digital en el buque. En primer lugar, y como consideraciones previas, se analizarán las características principales con las que debe contar un gemelo digital. A continuación, se definirá la arquitectura del gemelo digital: se establecerán los componentes elementales con los cuales debe contar y cómo estos estarán conectados entre sí.

Finalmente, se concretará la aplicación de esa definición de arquitectura para la aplicación de la metodología propuesta a un buque y qué criterios concretos será necesario valorar en este caso.

#### 3.1. DEFINICIÓN DE GEMELO DIGITAL

Como se adelantó en el capítulo 2, la definición más sencilla que se puede enunciar para un gemelo digital es la siguiente: “Un gemelo digital es una réplica digital de un activo físico, la cual mantiene una relación con el sistema real, de forma que entre ambos existe un intercambio de información (del sistema físico al virtual y viceversa).



**Imagen 3-1: Definición de gemelo digital**

No se entrará en este apartado en un exhaustivo análisis de las definiciones que se pueden encontrar en la bibliografía sobre el concepto de gemelo digital, ya que existen ya numerosas referencias que han llevado a cabo este tipo de revisiones (Semeraro et al., 2021; VanDerHorn & Mahadevan, 2021a, 2021b). Si bien, se debe destacar el hecho de que el concepto de gemelo digital es a

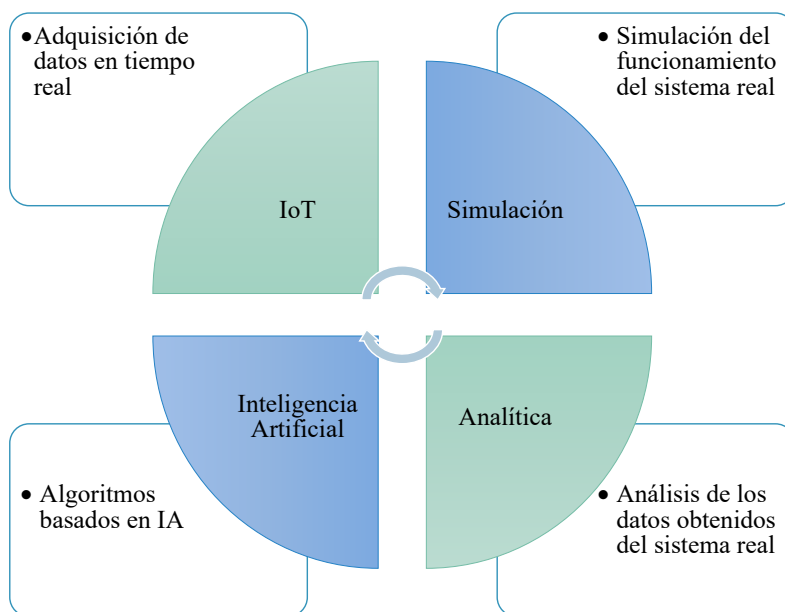
veces ambiguo, y lo que es más importante, tiene una definición variable en el tiempo. Desde los inicios del concepto como tal hasta hoy su definición ha ido evolucionando, y se espera que lo siga haciendo.

La definición de gemelo digital a menudo difiere si se enfoca desde el mundo industrial o desde el académico (Trauer et al., 2020), pero por norma general se define como como una representación (o un modelo) virtual que puede interactuar con el sistema real a lo largo de todo su ciclo de vida (Glaessgen & Stargel, 2012; M. Grieves & Vickers, 2017). En cambio, de acuerdo a otras definiciones, para poder considerar que nos encontramos frente a un gemelo digital es imprescindible que exista un intercambio de información que abarque sensores, datos y modelos (Lee et al., 2013; Negri et al., 2017). Según otra definición clásica se considera que un gemelo digital es la parte cibernética de un sistema ciberfísico (Alam & El Saddik, 2017; Graessler & Poehler, 2018).

Además de describir el concepto de gemelo digital, en el capítulo 2 se adelantó cuáles de las principales tecnologías habilitadoras, KETs, han hecho posible el despliegue de gemelos digitales, y que son:

- Internet de las Cosas, IoT.
- Simulación.
- Analítica.
- Inteligencia artificial.

El planteamiento que se propone en esta tesis abarcará el concepto de gemelo digital girando en torno a estos cuatro conceptos, es decir, se propondrá un gemelo digital que cuente con los siguientes componentes y funcionalidades, cada uno de ellos directamente relacionado con una de las tecnologías habilitadoras como se describe en la imagen 3-2:



**Imagen 3-2: Tecnologías habilitadoras y funcionalidades principales de un gemelo digital**

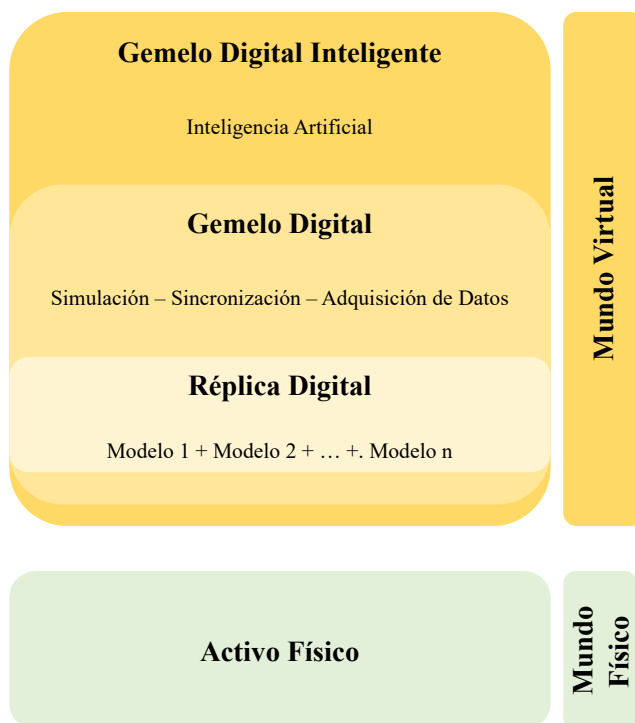
En este planteamiento, se puede afirmar que para considerar un gemelo digital de un activo físico (ya sea un producto, un sistema o un proceso) se deberá contar con lo siguiente:

- Sistema de adquisición de datos en tiempo real en el activo físico, formado por una red de sensores y/o actuadores inteligentes, que tengan capacidad de transferencia de información.
- Modelos de simulación que repliquen el funcionamiento del sistema real.
- Métodos de analítica, que permitan analizar los datos obtenidos del sistema real.
- Algoritmos de aprendizaje basados en inteligencia artificial que permitan el desarrollo de aplicaciones que optimicen el funcionamiento del sistema o producto.

No se trata de un planteamiento absoluto, sino que diverge bastante, dependiendo de la bibliografía consultada. A veces se habla de gemelo digital únicamente cuando se cuenta con una representación digital de la parte física (en ese caso sería más correcto emplear el concepto de maqueta virtual). En otras ocasiones, se denomina gemelo digital a la conexión entre la maqueta

virtual y la parte física a través del IoT, siendo sin embargo poco más que un sistema de monitorización.

Ashtari desagrega los gemelos digitales en 3 niveles desde un punto de vista de la complejidad de estos, es decir, dependiendo del grado de innovación que se alcanza cada uno de ellos, tal y como se muestra en la imagen 3-3 (Ashtari Talkhestani et al., 2019).



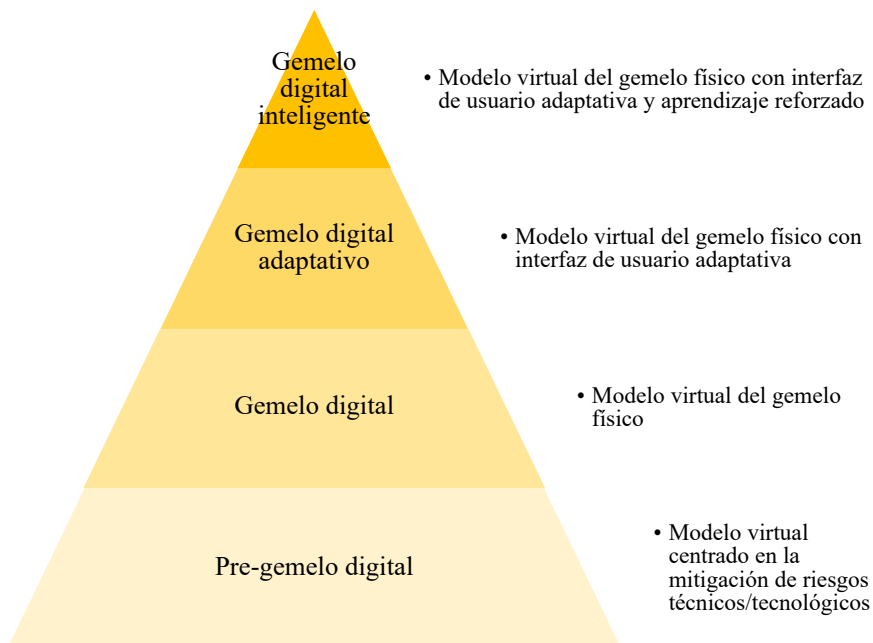
**Imagen 3-3: Niveles de gemelo digital según Ashtari**

En el nivel más básico, al que se denomina réplica digital, cada activo físico consta de sus modelos y datos. Estos modelos y datos combinados forman una copia virtual del activo. Si esta réplica digital está equipada con las siguientes tres características: sincronización con el activo físico, adquisición activa de datos y capacidad de simulación, la definición de réplica digital se transforma en gemelo digital. Finalmente, se dice que un gemelo digital inteligente, cuando además de contar con todas las características de un gemelo digital, este posee inteligencia artificial para realizar acciones de forma autónoma. Este gemelo digital inteligente puede, por lo tanto, implementar algoritmos de



aprendizaje automático, así como probar continuamente escenarios hipotéticos, allanando el camino para el mantenimiento predictivo y un funcionamiento general más flexible y eficiente.

Algo similar desarrolla Madni, aunque en este caso se presentan diferentes niveles en función del grado de madurez que presente el gemelo digital, de acuerdo a la imagen 3-4 (Madni et al., 2019).



**Imagen 3-4: Niveles (según grado de madurez) en el gemelo digital.** (Madni et al., 2019)

En el nivel más bajo se sitúa lo que se ha denominado tradicionalmente como prototipo virtual. Surge en las primeras etapas de la ingeniería y sirve de apoyo a la toma de decisiones durante el diseño conceptual y diseño preliminar. El prototipo virtual es un modelo del sistema físico previsto y que normalmente se crea antes de que se construya el prototipo físico. Un prototipo virtual se utiliza principalmente para validar ciertas decisiones clave sobre el sistema y mitigar riesgos técnicos específicos al principio del proceso de diseño.

El segundo nivel se corresponde con un gemelo digital en el que el modelo virtual es capaz de incorporar datos de rendimiento, estado y mantenimiento del gemelo físico. La representación virtual recibe actualizaciones del sistema

físico que permite actualizar el modelo virtual, y con ello se puede apoyar la toma de decisiones de alto nivel en diseño conceptual, preliminar y desarrollo.

Dado que la interacción con el sistema físico es bidireccional, existe también la posibilidad de que el gemelo físico utilice el conocimiento generado en el sistema virtual.

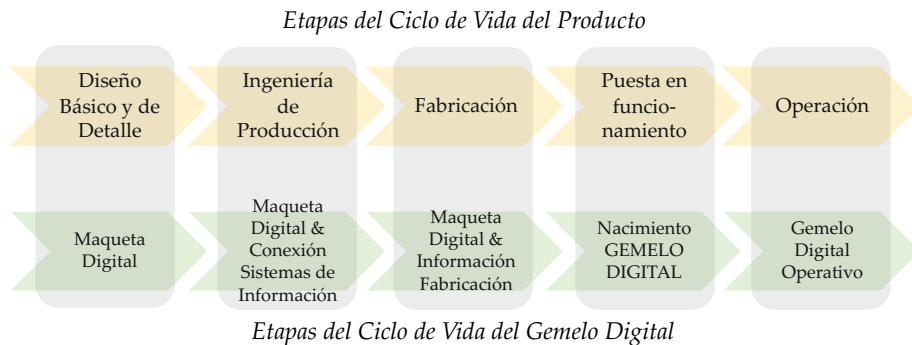
En el tercer nivel se define el gemelo digital adaptativo. Ofrece una interfaz de usuario adaptativa para los gemelos físicos y digitales. La interfaz de usuario es sensible a las preferencias y prioridades del usuario / operador. Una capacidad clave en este nivel es la capacidad de aprender las preferencias y prioridades de los operadores humanos en diferentes contextos. Los modelos empleados dentro de este gemelo digital se actualizan continuamente en función de los datos que se “extraen” del gemelo físico en tiempo real. Este gemelo digital puede respaldar la planificación y la toma de decisiones en tiempo real durante las operaciones, el mantenimiento y el soporte.

El cuarto nivel se correspondería con el gemelo digital inteligente. Tiene todas las capacidades de un gemelo digital del tercer nivel pero, además, tiene la capacidad de aprendizaje automático no supervisado para discernir objetos y patrones encontrados en el entorno operativo. Este gemelo digital en este nivel tiene un alto grado de autonomía.

### **3.2. ETAPAS PARA LA CREACIÓN DE UN GEMELO DIGITAL**

El nacimiento del gemelo digital puede surgir a lo largo del ciclo de vida del activo. En el caso de un producto, lo deseado es que nazca en las primeras fases y que vaya creciendo a medida que lo hace el desarrollo del producto.

Sin embargo, en el caso de un proceso puede suceder que el gemelo digital surja cuando el proceso está plenamente implementado en la realidad y que lo que se busque con el gemelo digital sea la optimización de ese proceso, la modificación de actividades, o el uso para un mejorado mantenimiento basado en el gemelo.



**Imagen 3-5: Esquema Ciclo de vida de un producto vs etapas creación gemelo digital**

En la **fase de diseño conceptual de un producto** se puede empezar a hablar de gemelo digital (o para ser más exactos del concepto de maqueta digital). Si se considera la fabricación de un sistema más o menos complejo, desde el momento en el que empiezan a aparecer los primeros diseños básicos de ese sistema, así como modelos de simulación elementales, se puede considerar que empieza a crecer el germen de lo que será el posterior gemelo digital.

Dentro de esta fase de diseño, cuando se empieza a profundizar en el diseño de detalle, aparecen los primeros modelos 3D. En esta etapa cobra gran importancia el desarrollo de modelos de simulación, los cuales permitirán llevar a cabo un análisis de alternativas de forma casi ilimitada, conociendo a priori cuál va a ser el comportamiento del producto. Algo clave en la toma de decisiones y para la optimización del diseño.

En la fase de **ingeniería de producción**, se da un paso más en el avance del gemelo digital, ya que en esta etapa lo que en principio era una maqueta virtual, formada por modelos CAD y de simulación, pasa a ser un modelo de fabricación, aparecen listas de materiales, secuencias de fabricación, órdenes de trabajo... Esta maqueta digital se conecta con los Sistemas de Información de la Empresa, principalmente con el Gestor del Ciclo de vida del producto (o, por sus siglas en inglés, Product Lifecycle Management, PLM), y con el sistema de planificación de recursos empresariales (Enterprise Resource Planning, ERP).

Durante la **fase de construcción**, se van incorporando a esta maqueta la información correspondiente al proceso de fabricación del producto. Esta

maqueta puede ser actualizada con las modificaciones que puedan surgir durante la fabricación. Además, comienzan a llevarse a cabo pruebas en el sistema real, que pueden alimentar la parte virtual, con lo cual se podría considerar casi un gemelo digital completo.

Una vez que el producto está finalizado y pasa a un estado operativo de **entrada en funcionamiento**, se puede afirmar que se trata de un gemelo digital de ese producto. La sensorica presente en ese producto o sistema, permite la recopilación de información del funcionamiento real, se produce la conexión digital/física. Incluso, si el gemelo ha sido planteado con esa funcionalidad, se puede alimentar al sistema físico a través del sistema digital (recomendaciones de funcionamiento en función de la información generada por el gemelo digital, etc.) A partir del momento en el que se empiezan a recoger datos de funcionamiento real (recopilación de **datos históricos**), es posible aprender de los mismos (analítica de datos).

Y, por último, se podrán llevar a cabo tareas de implementación de técnicas de **inteligencia artificial** que complementen ese análisis de datos y sirvan para la realización de tareas de análisis predictivo.

De lo anterior se deriva que las posibles aplicaciones de un gemelo digital dependen en cierta forma la etapa del ciclo de vida del producto que se está modelando. Cuando el gemelo digital surge al principio del ciclo de vida y va evolucionando pero sigue existiendo en cada una de las etapas se habla de entramado digital (Siemens, n.d.), ya que, como su propio nombre indica, el gemelo digital en este caso se entrelaza en todas las etapas de los ciclos de vida de los productos y acumula la información que se recopila en todas esas etapas.

### **3.3. CARACTERÍSTICAS INTRÍNSECAS DE UN GEMELO DIGITAL**

Erikstad define un gemelo digital como “un modelo digital capaz de representar el estado y el comportamiento de un activo real único en (casi) tiempo real”, identificando 5 características principales que debe cumplir un gemelo digital (Erikstad, 2018):

- **Identidad** (unicidad), en la conexión gemelo digital / gemelo físico. Cuando se observa un estado del gemelo, esta observación se corresponde únicamente con una realidad del sistema físico que representa
- Capacidad de **representación** gráfica del sistema físico real en un entorno digital (CAD, CAE).
- Capacidad de obtener, en tiempo real, el **estado** del sistema físico.
- Capacidad de reproducir el **comportamiento** del sistema real en respuesta a estímulos externos.
- Capacidad de describir el **contexto** en que este se encuentra, ya sean condiciones climatológicas, oceánicas, ...

Este planteamiento es similar al que presentan Cabos & Rostock, donde se proponen como características esenciales que debe contemplar un gemelo digital las siguientes (Christian Cabos & Rostock, 2018):

- **Representación de activos:** es decir, una representación digital de un objeto físico único.
- **Modelo de comportamiento:** lógica codificada para permitir predicciones y / o decisiones sobre el gemelo físico.
- **Datos de condición y configuración:** Datos que reflejan el estado y los cambios en el objeto físico único durante sus fases de ciclo de vida.

De modo que, a la hora de plantear el desarrollo de un gemelo digital, se ha de tener en cuenta que éste debe cumplir con lo siguiente:

En primer lugar, tiene que poder **caracterizar a un producto de forma única** y representarlo desde el punto de vista de su naturaleza física (eléctrica, mecánica, hidráulica...) y en algunos casos también desde el punto de su forma física. Además, debe poder representar en un entorno CAD el sistema físico que está reproduciendo y que debería de incluir los datos que se corresponden con cada uno de los componentes. Además de ello, aportará valor que el sistema, no incluya únicamente esta representación gráfica, si no que también pueda mantener una conexión con el gestor del ciclo de vida del producto, PLM, e incluso con el sistema de apoyo logístico del fabricante, de tal forma que se pueda mantener actualizado ese gemelo y acceder a funcionalidades y relacionadas con la logística del sistema. Esto permitirá asociar la información capturada de las fases de producción y/o de operación del objeto físico. Información que puede ir desde algo tan básico como la configuración del sistema,

es decir, los componentes que lo forman y la conexión que existe entre ellos. Pero que también podría abarcar, por ejemplo, un histórico de imágenes que se van tomando de un elemento (como podría ser el caso de una estructura de acero) para poder hacer un seguimiento su estado a lo largo del tiempo.

Resulta importante señalar, que el objeto físico solo necesitará ser modelado hasta el nivel de detalle necesario para el uso que se pretende de él. No siempre un modelo con más detalle implicará ventajas en cuanto al desarrollo del gemelo digital se refiere.

A esta representación de activos se le puede denominar modelo digital, teniendo en cuenta que no se está hablando de un modelo de simulación del sistema u objeto físico, sino de la representación de lo que ese sistema encarna, al que se le pueden asociar datos capturados (ya sean del diseño o datos hallados en la operación).

En segundo lugar, el gemelo digital deberá tener la capacidad de **reproducir el comportamiento del sistema físico**. Esta será una de las funcionalidades básicas y más relevantes de un gemelo digital. El gemelo deberá disponer de una de modelos que permitan simular el comportamiento del sistema físico y que además de ello permitan al usuario poder saber si el comportamiento del sistema físico se ajusta a lo que debería ser su funcionamiento normal, o si se está desviando del mismo.

Además, de forma ideal se valorará que los modelos matemáticos puedan ajustarse (o aprender) en tiempo real utilizando los datos de operación del sistema físico. La simulación mediante el uso de estos modelos podrá utilizarse para la realización de estimaciones o para predicciones de funcionamiento futuro, y también para el mantenimiento del sistema físico real. Para todo ello, se podrá plantear la utilización de algoritmos basados en inteligencia artificial como el machine learning.

Es importante señalar que, cuando se habla de modelo de comportamiento, en realidad se puede estar hablando de múltiples modelos, entre otros:

- Modelo de simulación de comportamiento en el dominio del tiempo.
- Modelo matemático para estudiar, por ejemplo, la degradación del casco y como afecta al rendimiento del buque.

- Modelo de elementos/volúmenes finitos que represente, por ejemplo: el casco o la hélice.
- Modelo complejo, que represente un sistema, y que incluya el control de este.

En algunos casos, estos modelos de simulación podrán ser actualizados a partir de los datos obtenidos del funcionamiento del sistema (observación). La representación del activo aquí es el enlace necesario para permitir la actualización automática o carga del modelo de simulación basado en mediciones u observaciones.

Por último, el gemelo digital debe poder **obtener toda la información necesaria para su funcionamiento**, ya sea información del propio sistema físico o información de contexto. La información del sistema físico será captada a partir de sensores desplegados en este, y que le deben permitir obtener su situación en tiempo real. Además, deberá almacenar la información operativa. En el caso de un buque, la ruta, condiciones ambientales, cargas externas a las que se pueda ver sometido. También deberá incorporar la información de aquellos procesos que se lleven a cabo en la fase de construcción que puedan ser de interés para el gemelo (por ejemplo, algún tratamiento que se haya aplicado a la estructura).

De nuevo, en este caso es imprescindible la representación del activo para que toda la información que se quiera incorporar pueda estar asociada al equivalente digital del objeto físico. De esta forma, estos datos se convierten en información útil. Algunos ejemplos son las mediciones de espesor para las cuales se registran la ubicación y la hora en el modelo. Los datos de medición se vuelven útiles al conectarlos a su parte estructural en el modelo. Otro ejemplo son las mediciones de vibración tomadas periódicamente en una caja de cambios. Si estos resultados de medición periódicos se asocian sistemáticamente a la representación del activo, se pueden detectar tendencias en el nivel de vibración y se puede planificar el mantenimiento en función de la condición real.

### **3.4. REQUISITOS DE IMPLEMENTACIÓN DE UN GEMELO DIGITAL**

Slot establece 3 requisitos generales que se deben de tener en cuenta a la hora de implementar un gemelo digital (Slot et al., 2020):

- **Flexibilidad:** Un gemelo digital es una representación digital de un activo físico que está en constante evolución. Por lo tanto, crear una instancia de un gemelo digital será un proceso que requiera de evaluaciones y ajustes a lo largo del tiempo, en muchos casos incluso empleando el método de “prueba y error”. Se debe de tener en consideración que los datos disponibles nunca serán los definitivos ni exactos, y nunca estarán lo suficientemente completos como para establecer de manera inequívoca el gemelo digital. Por lo tanto, este gemelo digital deberá de ser dinámico y flexible para poder añadir y eliminar características, flujos de datos, componentes de visualización, requisitos del usuario, etc. La creación de instancias para un gemelo digital es por lo tanto un proceso continuo que no necesita tener un punto estable final establecido. Es necesario adaptar el gemelo digital para que sea flexible, reconfigurable y escalable para posibles (y probables) cambios futuros.
- **Enfoque modular:** En lugar de considerar un gemelo digital como un modelo completo de una entidad física, este debe de verse como un “conjunto de conjuntos” de datos, modelos y simulaciones vinculados que evolucionan a lo largo del ciclo de vida del producto. De modo que, para cada instanciación de un gemelo digital, el contenido concreto puede variar, pero el gemelo digital debe proporcionar la forma en la que estos módulos interactúan entre sí. Esta estructura permite que estos módulos además puedan ser empleados en otras instancias de gemelos digitales. De esa forma, módulos individuales pasan a ser componentes funcionales que pueden contribuir en múltiples entornos.
- **Intercambiabilidad:** Cualquier subsistema, entidad, algoritmo o flujo de datos debe de poder verse como un módulo que puede ser reemplazado por otro módulo manteniendo (al menos en gran parte) la misma funcionalidad, mientras que tanto los mecanismos, enfoques e



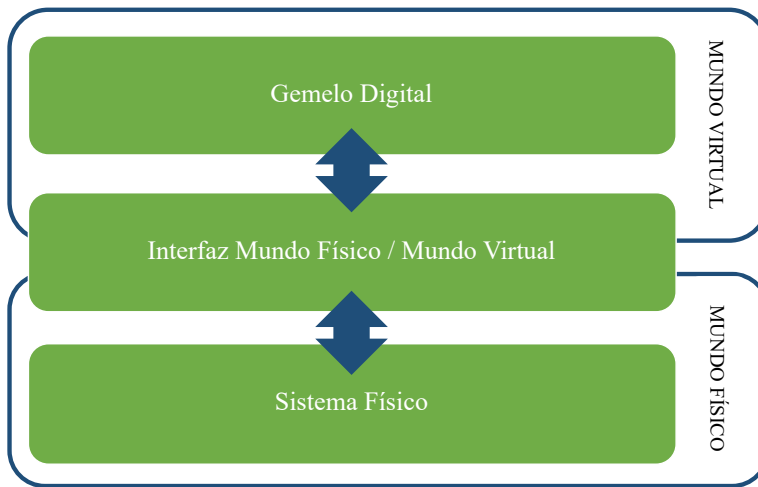
información que emplea el módulo, pueden diferir de forma significativa. De esta forma se puede emplear un repositorio de módulos predefinidos, así como módulos de caja negra. Estos últimos son bloques donde la funcionalidad y las interfaces se pueden definir, pero los métodos de trabajo y las capacidades de procesamiento de la información deben desarrollarse. Con el desarrollo de estos módulos (plug and play), se adquiere flexibilidad para agregar, reemplazar y eliminar módulos en gemelos digitales existentes o en aquellos que están en desarrollo, permitiendo hacer avances de forma rápida y fiable, con módulos que ya han sido previamente probados en otro contexto.

### **3.5. ARQUITECTURA GENERAL PARA IMPLEMENTACIÓN DE UN GEMELO DIGITAL**

Como se ha comentado, en el caso objeto de estudio, el gemelo digital es un sistema que tiene que ser capaz no sólo de reproducir el funcionamiento de su gemelo físico, sino que también debe permitir analizar la bondad y salud de su funcionamiento, debe proponer posibles correcciones o consejos de operación a los usuarios y adicionalmente debe poder auto-ajustarse y generar simulaciones de operación de modo independiente al sistema físico.

Para el desarrollo del gemelo digital existen cuatro tecnologías facilitadoras que sirven como principal soporte: IoT, simulación, analítica e inteligencia artificial, que serán las que permita realizar la conexión entre el mundo físico y el virtual y dotarán al gemelo de las funcionalidades requeridas.

El esquema más general al que se puede hacer referencia es aquel que relaciona el “gemelo físico” con el “gemelo virtual”, como el que se muestra en la imagen 3-6.



**Imagen 3-6: Componentes principales para la implantación de un gemelo digital**

En el nivel más bajo se encuentra el nivel físico compuesto por el sistema real. En la capa intermedia estaría la interfaz entre el mundo físico y el virtual, la cual está formada por los sensores y PLCs del sistema. En el nivel superior se encuentra lo que se conoce propiamente como gemelo digital, es decir, la parte virtual del gemelo mediante la cual se integrará la visualización, el almacenamiento y el procesamiento de información de todo el sistema físico y virtual.

Siguiendo estos condicionantes, la arquitectura de un gemelo digital debería estar formada por los componentes necesarios que integren lo que a continuación se enumera. Esta descomposición, o similares, de componentes del gemelo digital, aparece en la bibliografía consultada (Harper et al., 2019; Malakuti et al., 2020), y es la que toma como referencia el Industrial Internet Consortium:

- Sistema de adquisición de datos: que recopile todas las señales necesarias del sistema físico. Un sistema de toma de datos que sensorice todos los datos necesarios del sistema físico.
- Modelos de datos, o modelos de visualización y de simulación derivados de la caracterización del sistema.
- Algoritmos que permitan a partir de los datos de las simulaciones, y de los obtenidos del sistema físico, establecer rutinas de control, mantenimiento y ajuste.

- Almacenamiento de datos, en que se puedan guardar tanto los datos sensorizados como los generados por el propio gemelo.
- Inteligencia artificial y machine learning: algoritmos que generen las posibles estimaciones, predicciones, detección de anomalías, etc., a partir del análisis de los datos generados por el gemelo y las señales del sistema físico y los datos del histórico.
- Y finalmente, una unidad de visualización e interfaz de usuario (HMI), que permita la interrelación entre el gemelo y el cliente final del mismo.

A continuación, se describirán de forma general los principales componentes que forman parte de la arquitectura del gemelo que se acaba de definir.

### 3.5.1. Sistema físico

En la capa más baja de la arquitectura se encuentra el sistema físico. Puede tratarse de una máquina, un sistema o un conjunto de sistemas más complejo, dependiendo de la entidad que se esté considerando a la hora de plantear el gemelo digital. Por ejemplo, el sistema físico de un gemelo digital de una bomba para circulación de combustible sería únicamente la bomba, pero si se considera un sistema de distribución de combustible, el sistema físico estaría formado por la bomba de distribución, tuberías, válvulas y demás componentes. Finalmente, si se considera un vehículo, el sistema físico estaría formado por el chasis, el sistema de combustible, el motor, el sistema de frenado, etc.

La forma en que el sistema real aporta valor al gemelo digital será en forma de información. Esta información será generada por los sensores del sistema (internos o externos) así como de parámetros de funcionamiento (que deberán de ser transmitidos a las capas superiores de igual forma que los datos recogidos por sensores por lo que, en términos de gemelo digital, son equivalentes). Un sensor, es aquel dispositivo que permite medir una magnitud física (o química): temperatura, humedad, distancia, inclinación, velocidad... y transferirla en una variable eléctrica, de forma que esta puede ser transmitida. En función de los datos de salida, se pueden clasificar en analógicos o digitales:

Existen diferentes tipos de sensores, en función del tipo de variable que tengan que medir o detectar:

- De contacto.
- Ópticos.

- Térmicos.
- De humedad.
- Magnéticos.
- De infrarrojos.
- Otros.

Tradicionalmente se han empleado sensores cableados, aunque se ha de señalar que cada vez más, se empiezan a ver en entornos industriales sensores con comunicación inalámbrica.

Los datos recogidos por los sensores se transmiten a las capas superiores a través de PLCs (RTUs). Un PLC o autómatas programables, es un pequeño ordenador empleado en procesos industriales, que es capaz de comunicarse directamente con los equipos que forman el sistema y de actuar sobre ellos (dar órdenes, recibir información...). Desde que aparecieron, en los años 70, han formado parte de los procesos de automatización industrial. Si bien, en los últimos años, han ido evolucionado, en línea a las transformaciones digitales que se están produciendo en la industria, aumentando su capacidad de cálculo, velocidades en comunicación e incorporando características 4.0.

La comunicación de campo, entre los sensores o actuadores y PLCs puede tener diferente naturaleza y corresponder con protocolos de comunicaciones como son: lazo de corriente (comunicación entre el sensor físico y el transmisor de ese sensor, la típica señal de corriente eléctrica de 4-20 mA), Hart, Fieldbus, Profibus, Profinet, Modbus.

En el entorno del gemelo digital y del IoT en general, se busca la capacidad de disponer de una red/protocolo de comunicaciones que permita:

- Rapidez en la Comunicación.
- Transferir grandes volúmenes de datos.
- Escalabilidad.
- Fiabilidad.
- etc.

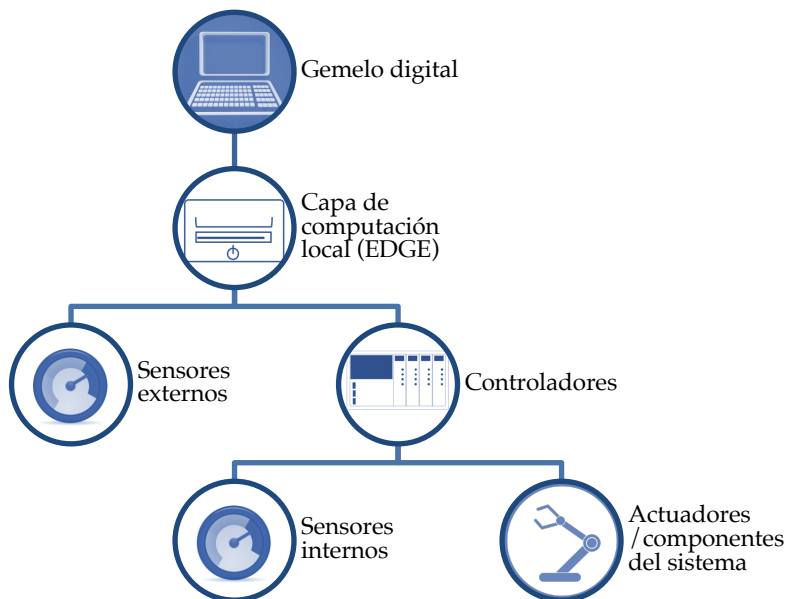
Es por ello, que a pesar de que se mantienen activos los protocolos MODBUS y PROFIBUS, cada vez está adquiriendo más fuerza el protocolo PROFINET, un estándar de comunicaciones para ethernet industrial basado en el protocolo TCP/IP e IT y sucesor de PROFIBUS (bus de campo en serie clásico,

basado en RS-485), con el que mantiene alguna similitud y con el que permite integrarse.

### 3.5.2. Interfaz mundo físico / mundo virtual

La interfaz entre el mundo físico y el mundo digital estará compuesta principalmente por el sistema de adquisición de datos que fluye desde el sistema físico hacia el gemelo virtual. Pero, además, en aquellos casos en los que se contemple que el gemelo podrá actuar (tomar decisiones) sobre el sistema físico, el traslado de órdenes desde el mundo virtual al físico también se producirá a través de esta interfaz.

El sistema de adquisición de datos del sistema real, parte de la capa física, en las que se encuentran los componentes del sistema, así como los sensores y controladores que este incorpore, y que serán los encargados de recopilar información del funcionamiento del sistema. Además de ello, como se acaba de comentar, cabe la posibilidad de que el gemelo actúe sobre el sistema físico, por lo que será esta capa la encargada de comunicar las acciones indicadas por el gemelo a los actuadores del sistema físico.



**Imagen 3-7: Componentes de capa física e interfaz hacia el gemelo digital**

A menudo, sobre la capa física se sitúa una capa denominada EDGE. El edge computing (Knebel et al., 2020) es una tecnología que permite capturar,

procesar y analizar datos en tiempo “casi real”. Se presenta en contraposición al Cloud Computing, donde los datos se envían a las capas superiores sin ningún tipo de procesado, lo cual a veces implica, el trasvase de enormes volúmenes de datos que a veces no son útiles.

Las principales ventajas que aporta disponer de esta capa edge:

- Análisis de datos en tiempo real.
- Reducir latencias.
- Consumir menos ancho de banda.
- Rapidez.
- Disminución de cuellos de botella en el tráfico de datos.
- Incremento de la seguridad.

Eso no significa que la dependencia de la nube y de entornos cloud computing desaparezca: ambas tendencias deben aportar, y por ejemplo edge computing es más adecuado cuando sobre todo se necesita velocidad y baja latencia en esas transferencias de datos, mientras que la nube seguirá siendo protagonista para analizar y tratar grandes cantidades de datos que requieren una potencia de cálculo notable.

En general, la comunicación que parta hacia la capa edge puede emplear tanto bus de campo como protocolos más avanzados. Y aquello que parta hacia las capas superiores empleará: MQTT, API Rest o, hacia lo que parece que se encamina la tecnología IoT, OPC UA.

El protocolo MQTT es un protocolo de comunicación Machine to Machine (M2M) empleado habitualmente para la comunicación de dispositivos IoT (Liu et al., 2020; Patel & Doshi, 2020). Se trata de un protocolo sencillo y ligero, dos de las características que justifican su extendido uso en desarrollos de este tipo, donde muchas veces existen limitaciones importantes relacionadas con la potencia, el consumo y/o el ancho de banda.

Una API REST define un conjunto de funciones, reglas y especificaciones que las aplicaciones pueden emplear para comunicarse entre ellas mediante peticiones HTTP, con las que se pueden realizar solicitudes y recibir respuestas, como: *get*, *post*, *put* y *delete* (Breu et al., 2008; Gallegos Valera, 2013; Kumar Deepak, 2018).

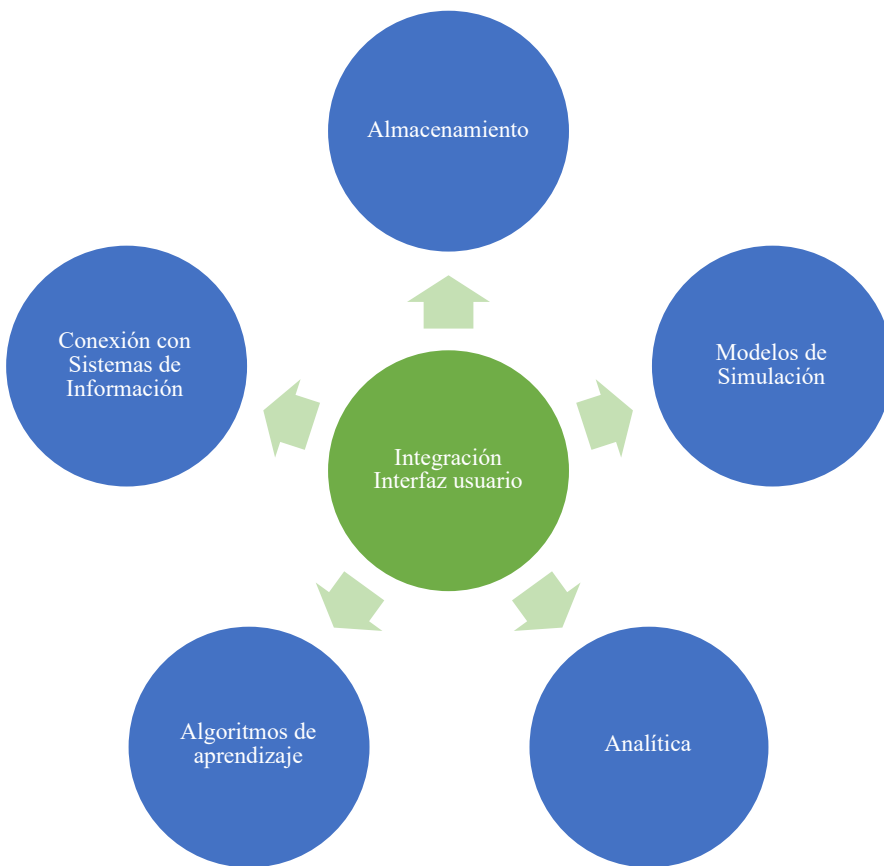
El protocolo OPC, Open Protocol Communication es, como indica su nombre, un protocolo de comunicación abierto (e independiente) de aplicación industrial, y que se encuentra habitualmente en software industrial como los sistemas SCADA, historiadores, drivers, etc. (González et al., 2019)

La tecnología OPC se basa en una arquitectura cliente/servidor en la cual existe un servidor que se conecta a la fuente de datos y un cliente que se encarga de realizar las consultas para hacer uso de esos datos. Este sería el caso de aquellos servidores que están conectados a PLC's y que se comunican con un cliente, como podría ser un sistema de tipo SCADA. Las principales ventajas que ofrece el protocolo OPC son: en primer lugar, se trata de una tecnología escalable, en segundo lugar, se trata de un protocolo independiente de los fabricantes y además su mantenimiento es relativamente sencillo. En cambio, presenta importantes inconvenientes como son su dependencia del SO Windows, y su debilidad en seguridad. Ambas deficiencias se han visto superadas con el desarrollo del nuevo protocolo que se ha denominado OPC UA (Burke, 2016)

### 3.5.3. Gemelo digital

En este apartado se definirán los componentes que conforman la parte virtual del gemelo. Se puede afirmar que una plataforma de gemelo digital deberá contener, como mínimo (tal y como se representa en la imagen 3-8), lo siguiente:

- Integración de componentes del gemelo. Interfaz de usuario, que permita la monitorización del sistema real, la visualización de modelos 3D, el acceso a documentación, etc.
- Almacenamiento de datos (reales y simulados).
- Modelos de simulación: en tiempo real o para predicción de comportamiento.
- Análítica de datos.
- Algoritmos de aprendizaje.
- Integración/conexión con los sistemas de información.



**Imagen 3-8: Componentes de un gemelo digital (Mundo virtual)**

### 3.5.3.1. Integración de funcionalidades. Interfaz de Usuario

Es evidente que el Gemelo deberá contar con una interfaz de usuario, que permita el acceso a la información que este contiene. Las funcionalidades a las que se pueda acceder dependerán de lo que se abarque en el proyecto concreto, pero en general, deberían de ser algo similar a lo siguiente:

- Representación de un mímico del sistema: modelo simplificado del sistema y visualización de señales en tiempo real.
- Representación de indicadores de relevancia: horas de funcionamiento, horas hasta próxima tarea programada (o estimada de



mantenimiento), alertas de funcionamiento (basadas en umbrales o en modelos de predicción), etc.

- Acceso a los datos de sensores (en tiempo real e históricos).
- Acceso a los modelos de simulación del sistema (en tiempo real, históricos, o predicción de comportamiento a futuro).
- Acceso a Modelos 3 D del sistema.
- Acceso a documentación.
- Acceso a modelos de aprendizaje (IA).
- Acceso a mapas y diagramas de flujo de información del gemelo digital.
- Realidad aumentada.

### 3.5.3.2. Modelos de simulación

Un parte indispensable de la arquitectura del gemelo digital son los modelos de simulación que describen de forma analítica cómo se comporta un sistema y permiten conocer qué tipo de respuestas se pueden dar frente a determinadas situaciones, pudiendo actuar antes de la aparición de fallos o funcionamientos incorrectos (detección temprana – detección de tendencias no deseables – árboles de fallos de la máquina).

Los gemelos digitales generan valor sobre un producto a lo largo de su ciclo de vida. Permiten aprovechar los datos generados por el IoT (datos reales) y por los modelos de simulación (datos simulados) para informar sobre el funcionamiento de un sistema y al mismo tiempo predecir mejor su comportamiento.

En muchos casos, los gemelos digitales que ofrecen análisis y mantenimiento predictivo se basan únicamente en datos históricos y modelos estadísticos (Coraddu et al., 2019; He et al., 2019). Esto presenta un problema: no se puede aplicar en las primeras fases de desarrollo de un producto o de puesta en marcha de un sistema, ya que se tarda tiempo en obtener datos que permitan realizar este tipo de análisis. Además, esos datos, deben de ser procesados y pasados a un formato utilizable. Se puede afirmar que el principal reto a la hora de emplear gemelos digitales basados en datos históricos radica en conseguir una cantidad suficiente de datos que permitan cubrir todos los escenarios posibles a los que podrá enfrentarse el producto en la vida real.

Plantear gemelos digitales que en lugar de sustentarse en modelos estadísticos en base a históricos lo hagan en modelos de simulación basados en física presenta importantes ventajas. En primer lugar, permiten complementar o incluso reemplazar datos históricos por datos obtenidos en simulaciones. Esto reducirá el número de pruebas físicas necesarias para la recopilación de una cantidad suficiente de datos, así como el tiempo de procesado de esos datos. Es más, la capacidad de un gemelo digital basado en la física para simular las condiciones de un sistema puede incluso reducir la cantidad de sensores necesarios en el dispositivo IoT.

En este caso, en lugar de tener que esperar a disponer de suficientes datos de campo para construir e implementar un gemelo digital, únicamente se necesita una pequeña cantidad de datos para validarlo. Esto acelerará el desarrollo y el tiempo de comercialización. Además, en muchas ocasiones se da el caso de que los productos del mundo real tienen un número limitado de sensores, e incluso algunos componentes no pueden tener sensores adecuados debido al costo u otras limitaciones. En estos casos una réplica virtual en tiempo real proporciona información y valor al combinar datos conocidos con simulación para "llenar los vacíos". Se tendrá entonces un número ilimitado de "sensores virtuales" proporcionados por la simulación. Los componentes pueden someterse a monitoreo, diagnóstico y se pueden realizar predicciones futuras, operadas de forma remota mediante un gemelo digital (Tiainen et al., 2019; Tiwana, 2019).

Ha de considerarse la posibilidad de combinar ambas tipologías de modelos en el desarrollo del gemelo digital. Dependiendo de la naturaleza del activo que se quiera simular, emplear modelos híbridos que estén basados en física pero que también incorporen modelos basados en datos estadísticos, permitirá alcanzar desarrollos que permitan simular el comportamiento real del sistema considerado con un elevado grado de fiabilidad (P. Wang & Luo, 2021).

En general, asociados al concepto de gemelo digital, se suele hablar de dos tipos de modelos de simulación:

**Modelos de predicción de comportamiento** en los cuales se puede ejecutar una simulación en un espacio asumible de tiempo, y que permita simular el comportamiento del sistema en un tiempo mayor. Este tipo de modelos permiten:

- Adquirir experiencia y conocimiento sobre el funcionamiento del sistema que se está simulando.
- Identificar problemas en un proceso complejo, en que empleando únicamente la experiencia o el conocimiento experto no es suficiente.
- Plantear escenarios alternativos en un sistema.
- Conocer el comportamiento de sistemas complejos, para los cuales fabricar siquiera un prototipo no sería abordable ni económica y tecnológicamente.

Además, este tipo de modelos pueden aplicados en cualquier fase del ciclo de vida del sistema que se está considerando, desde el diseño, construcción y operación.

**Modelos para simulación en tiempo real**, los cuales permitirán comparar en tiempo real las señales de los sensores con su equivalente simulado, además permitirá conocer variables para las cuales no se dispone de sensores o que su coste supone una importante limitación a la hora de ser instalados en un sistema.

Puede que sea de interés simular todo el sistema, o alguna parte crítica del mismo. Además, en algunos casos, será relevante emplear modelos de simulación complejos, como puede ser el caso de CFDs o en otros casos, el uso de sencillos modelos numéricos puede ser suficiente.

#### *3.5.3.2.1. Herramientas para generación de modelos de simulación de sistemas*

Este tipo de herramientas permiten desarrollar modelos 1D multidominio, que buscan simular el comportamiento de un sistema, y que son capaces de cubrir diferentes disciplinas (modelos multifísicos) y su integración entre ellas. Pueden incorporar, además, la simulación del control del sistema.

Dentro de estos tipos de simuladores se pueden citar: MATLAB SIMULINK, AMESIM, o por ejemplo aquellas herramientas que se basan en el lenguaje Modelica (OpenModelica, Dymola, Maplesim). Estas herramientas permitirán, en el entorno del desarrollo de un gemelo digital, simular sistemas de forma más o menos simplificados a partir de esquemas funcionales y/o especificaciones de los equipos que componen el sistema real.

### 3.5.3.2.2. Herramientas para simulación de sistemas complejos

Se incluyen en esta categoría herramientas de simulación que se emplean para una disciplina concreta, por ejemplo: FEM (Modelado por elementos finitos), CFD (Dinámica de fluidos computacional), MBS (Simulación de sistemas multi-cuerpo), etc. Este tipo de software permite adquirir una precisión mucho mayor que en el caso de los primeros. Pero en contra, requieren tiempo para llevar a cabo la simulación y, en algunos casos, ordenadores con elevadas prestaciones tanto de proceso como de almacenamiento.

Además, desarrollar un modelo con alta fiabilidad requiere un conocimiento detallado del sistema, el cual no siempre existe, sobre todo en las fases iniciales. Esto hace que este tipo de modelos no sean directamente utilizables para su uso en simulaciones de funcionamiento/monitoreo en tiempo real o para su uso en mantenimiento predictivo.

Una posible solución a estos inconvenientes, es el uso de Modelos de Orden Reducido, ROM (Hartmann et al., 2017): Un ROM es una versión simplificada de un modelo computacional de alta fidelidad que preserva el comportamiento esencial y los efectos dominantes mientras reduce el tiempo de solución y la capacidad de almacenamiento computacional requerida. Estos modelos son simplificaciones de modelos complejos, que permiten mantener la fiabilidad de su comportamiento. Permitiendo su uso en un entorno de gemelo digital, donde los tiempos de simulación deben de ser muy cortos.

### 3.5.3.2.3. Integración de modelos de simulación

Una vez desarrollados los modelos de simulación estos deben de poder ser ejecutados en la plataforma en la que se integre el gemelo digital. Además, es razonable pensar que a la hora de implementar los modelos de simulación de los sistemas que componen la entidad a la que se quiere aplicar el concepto de gemelo digital, habrá que integrar modelos de simulación de componentes que pueden haber sido desarrollados en diversas plataformas, como pueden ser las que se han citado en los apartados anteriores. En los últimos años se ha estado trabajando en un estándar que permita llevar a cabo esta integración, manteniendo la calidad de los modelos.

Consecuencia de lo anterior es el desarrollo del Estándar FMI: Functional Mockup Interface. (Bertsch et al., 2014) . El estándar FMI permite el

intercambio de modelos de simulación entre diferentes plataformas, además de la co-simulación de diferentes modelos (independientemente de la herramienta con la que hayan sido desarrollados). Este estándar fue desarrollado por ITEA (Information Technology for European Advancement) en el proyecto MODELISAR y que se centraba en el estudio de intercambios de modelos de simulación de sistemas multifísicos en el sector de la automoción (Chombart, 2012). Finalizado el proyecto, el desarrollo del estándar siguió adelante, creándose una entidad propia relativa al FMI.

Hoy en día, las principales herramientas de simulación permiten la importación y/o exportación de modelos basados en FMI, que se denominan FMU (Functional Mockup Unit). Algunas de las plataformas para simulación que permiten exportación de modelos FMU, son (Junghanns & Blochwitz, 2018): 20-sim, Adams, AMESim, ANSYS, CATIA, Dymola, EcosimPro, MapleSim, OpenModelica, Scilab, SimulationX, Simulink, etc.

Un fichero FMU es, en realidad, un fichero comprimido que contiene (Negri et al., 2019):

- Fichero xml en el que se almacenan las variables.
- Código C correspondiente con las ecuaciones.
- Otros datos, como tablas o comentarios.

La ejecución de estos FMU se puede hacer elaborando código propio, o empleando alguna de las herramientas y librerías ya desarrolladas con esta finalidad. Existen librerías para lenguajes C/C++ o para Python ampliamente probadas en la ejecución de FMUs y que se pueden implementar en una plataforma para gemelo digital (como puede ser una plataforma IoT).

También existen herramientas y software específicos para este tipo de aplicaciones: Livetwin de Siemens, Rexcore de Rexygen, ETAS de Labcar, dSPACE... que en general están pensados para la ejecución de modelos en tiempo real para soporte de gemelos digitales.

La ejecución de los modelos FMU en el entorno del gemelo digital se puede realizar, principalmente desde 2 perspectivas:

- Edge computing.
- Cloud computing.

Las ventajas de la primera se centran en la reducción de la latencia, ya que las señales reales solo han de viajar de la capa física a la capa de edge, y los modelos de simulación se ejecutan en esa misma capa. Por otro lado, las ventajas de ejecutar los modelos en la plataforma de IoT en la nube, hace que se pueda disponer de una mayor capacidad de cálculo, ya que, en el caso de modelos complejos, una plataforma edge, puede resultar algo justa (aunque para solventar esta limitación, aunque existiría la posibilidad de creación de modelos de orden reducido, ROM. Además, en este caso no sería necesaria la instalación de software en la capa edge, quedando todo centralizado en la nube.

### **3.5.3.3. Almacenamiento de datos**

El gemelo digital deberá tener capacidad para almacenar multitud de datos, a menudo heterogéneos: en todo momento se estarán recibiendo los datos provenientes del sistema de adquisición del sistema real, pero también de los modelos de simulación. Y no solo ello, sino también se podrán almacenar modelos CAD (2D o 3D) fotografías o videos del sistema o documentación como manuales de fabricante, de mantenimiento de equipos, etc.

Dependiendo de la plataforma, el tipo de almacenamiento varía desde bases de datos relacionales, orientadas a objetos, gráficas, etc. Será, por tanto, necesario definir el tipo de base de datos más adecuada, la capacidad máxima de almacenamiento y qué información se va a almacenar para conformar el grupo de datos históricos.

En cuanto a las diferentes alternativas de bases de datos se han de realizar los siguientes comentarios:

- Las bases de datos de tipo relacional (SQL) no son las más idóneas si lo que se busca es escalar datos de baja dimensionalidad (como puede ser el caso de tiempo-valor), hacia un número elevado de entradas. En este caso, lo recomendable es emplear bases de datos de tipo NoSQL, como: Redis, MongoDB o Apache Cassandra. En este caso, las bases de datos almacenan datos de tipo clave-valor obteniendo unos tiempos de respuesta mucho más rápidos que en el caso de las SQL.
- Además, existe un tipo concreto de bases de datos, dentro de las NoSQL, para el almacenamiento de series de tiempo y que consigue optimizar el almacenamiento de valores de sensores en el tiempo.

Además, estas bases de datos deben permitir insertar grandes cantidades de datos por unidad de tiempo, lo que hace que no se generen cuellos de botella en este punto. InfluxDB, KairosDB o Prometheus de código abierto o eXtremeDB o kdb+ de código propietario, serían algunos ejemplos de este tipo de bases de datos.

- En aquellos casos en los que sea un requisito mantener compatibilidad con otros programas que ya utilicen SQL, existen alternativas como TimescaleDB, que utiliza PostgreSQL en su interior y mantiene la compatibilidad con SQL, y al mismo tiempo, presenta las ventajas de las bases de datos para series de tiempo.

#### **3.5.3.4. Análisis de Datos y Algoritmos de aprendizaje y predicción**

Otro componente esencial del gemelo digital son los algoritmos de aprendizaje y predicción, a través de los que se pueden procesar los datos tanto del sistema físico como de los modelos de simulación y ajustar de forma inteligente algunos parámetros del gemelo. La potencia de los algoritmos de analítica es que permiten extraer patrones que a simple vista no sería posible, ya que la relación entre los datos es muy compleja. También se pueden predecir situaciones no deseables y actuar con mayor antelación.

#### **3.5.3.5. Conexión con Sistemas de Información y Logística**

Como se ha comentado en el al principio del capítulo, se puede afirmar que el gemelo digital surge (aunque no sea un gemelo digital totalmente funcional) en el momento en el que se inicia el diseño del sistema físico, ya que los primeros modelos CAD y modelos de simulación iniciales pueden ser el germen del futuro gemelo digital. En este momento, las listas de materiales (“Bill of Materials”, BOM) y modelos CAD (funcionales) se enlazan en el gestor del ciclo de vida del producto (PLM), el cual puede alimentar directamente al gemelo con la información disponible del producto en cada fase.

Una vez que el producto pasa a la fase de construcción, datos contenidos en el PLM pasan al Enterprise Resource Planning – Planificación de Recursos Empresariales (ERP), y es en el ERP donde se indica qué está hecho, qué falta por hacer, qué es necesario comprar, etc. Información logística, que puede ser de interés también en el gemelo digital.

Finalmente, está el Enterprise Asset Management – Gestor de Activos Empresariales (EAM), que se emplea para hacer seguimiento y gestión de un

activo a través de su ciclo de vida (construcción, puesta en servicio, operación, mantenimiento, reemplazo o desmantelamiento). Esta conexión entre los diferentes sistemas es lo que se denomina hilo digital, o digital thread (Pang et al., 2021), pero puede estar materializado a través del gemelo.

Integrar la documentación y modelos relativos al producto a través de los Sistemas de Información existentes, presenta importantes ventajas:

- Se asegura que la información y modelos está siempre actualizada.
- Se evita duplicidad de información.

Como principal inconveniente se puede dar que la comunicación entre el gemelo digital y los Sistemas de Información no siempre es sencilla, y puede requerir de programación en ambos lados de la conexión.

Por lo que se debe plantear como alternativa que sea el gemelo digital quien incluya directamente la documentación relativa al producto. Desde información relativa a la fabricación hasta manuales de mantenimiento. Si se realiza en el entorno del gemelo digital será de vital importancia la correcta estructuración de toda la documentación, además de un correcto control de versiones (por ello, si es posible, la integración con el gestor del ciclo de vida del producto y con el gestor de activos empresariales proporcionará importantes ventajas y ahorros en el despliegue del gemelo digital).

Igual que sucede con la documentación, la forma ideal de integrar los modelos 3D en el gemelo digital debería ser a través del PLM, de forma que se recojan todos los cambios que se vayan produciendo. De no ser posible, la plataforma de integración empleada para el gemelo digital deberá contar con capacidad para visualización de modelos 3D.

### **3.6. GEMELO DIGITAL EN BUQUES**

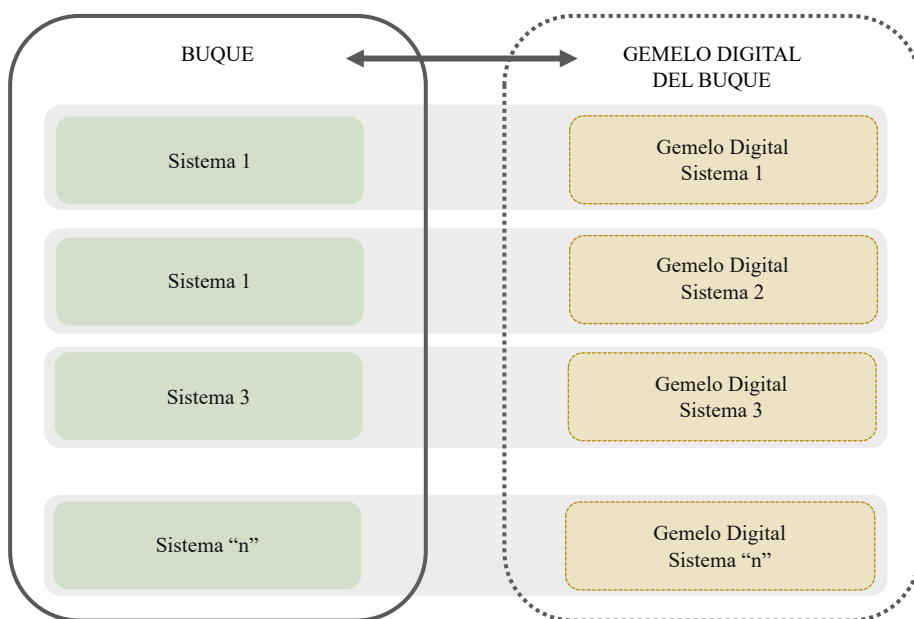
Se puede considerar un buque como un sistema ciberfísico formado por diferentes sistemas individuales, por lo tanto, un gemelo digital completo estará formado, a su vez, por otros gemelos digitales de menor dimensión, que corresponden con los sistemas individuales que forman el buque.

Grievés y Vickers hacen una distinción entre instancias de gemelos digitales y agregados (Michael Grievés & Vickers, 2016). Cuando se habla de instancia de gemelo digital (DTI) se refiere a aquel caso en el que el gemelo físico que



corresponde y permanece unido al gemelo virtual durante toda su vida útil. Un agregado de gemelos digitales (DTA) es la agregación de algunos DTI o incluso a su vez de otros DTA. Mientras que el DTI puede ser una estructura independiente, un DTA no. Por tanto, los DTI pueden ser interrogados por un DTA para conocer su estado actual del sistema.

De forma que se puede plantear el gemelo digital del Buque como un agregado de gemelos digitales, formado por otros agregados o instancias de gemelos digitales, que se corresponderían con los sistemas del buque.



**Imagen 3-9: Planteamiento esquemático para gemelo digital del buque desde el punto de vista del buque como agregado de sistemas**

A la hora de desplegar un gemelo digital de un sistema o grupo de sistemas físicos a bordo del buque, existen una serie de decisiones que es necesario adoptar, referentes a su arquitectura y, a partir de ellas, pueden establecerse una serie de necesidades y requisitos que deben cumplirse para su desarrollo.

### 3.6.1. Sistema físico

Como se ha comentado, la forma de abarcar la implementación del gemelo digital en el buque pasa por la generación e integración de diferentes gemelos

digitales, correspondientes a los sistemas que componen el buque, y que en general podrían ser:

- Sistema de propulsión y gobierno.
- Sistema de generación y distribución de energía.
- Sistemas auxiliares.
- Sistemas de control de daños.
- Sistema de alarmas.
- Sistemas específicos del buque: buque de carga, buque de pesca, buque de guerra, etc.

### **3.6.2. Interfaz mundo físico / mundo virtual**

De acuerdo con la información a la que se ha podido acceder, se puede estimar que hoy en día, un buque convencional puede integrar un elevado número de señales, que pueden estar entre las 2000, en el caso de un remolcador (DAMEN, 2019), pero que en el caso de un buque militar se puede llegar hasta las 30.000 como valor medio (García González, 2018), si se tienen en cuenta unidades de adquisición, subestaciones basadas en autómatas programables y puestos de operador. Se estima que los nuevos buques militares puedan llegar a las 50.000.

Teniendo en cuenta esta gran cantidad de señales que deberá gestionar el buque, será necesario estudiar la forma más eficiente a la hora transferir esta información. En el caso de aquellos buques que cuenten con un Sistema de Control, puede considerarse la alternativa de plantear un gemelo digital que se comunique con el Sistema de Control para obtener las señales necesarias, y que sólo en casos determinados y muy específicos, maneje sus propias señales y sensores. Evidentemente, uno de los retos de un proyecto como el que se plantea, reside en la definición y el planteamiento de la interacción entre estos dos sistemas embarcados.

En lo referente a los requisitos a aplicar para las comunicaciones se deberá tener en cuenta lo siguiente:

- Se empleará un estándar de ethernet industrial, como puede ser PRO-FINET (OMRON, 2012; Siemens, 2012) o, por ejemplo, otro tipo de estándares como los Time-Sensitive Networking (TSN) muy empleados

para la transmisión de datos con baja latencia y en tiempo real (Astarloa & Rodríguez, 2018; CISCO, 2017).

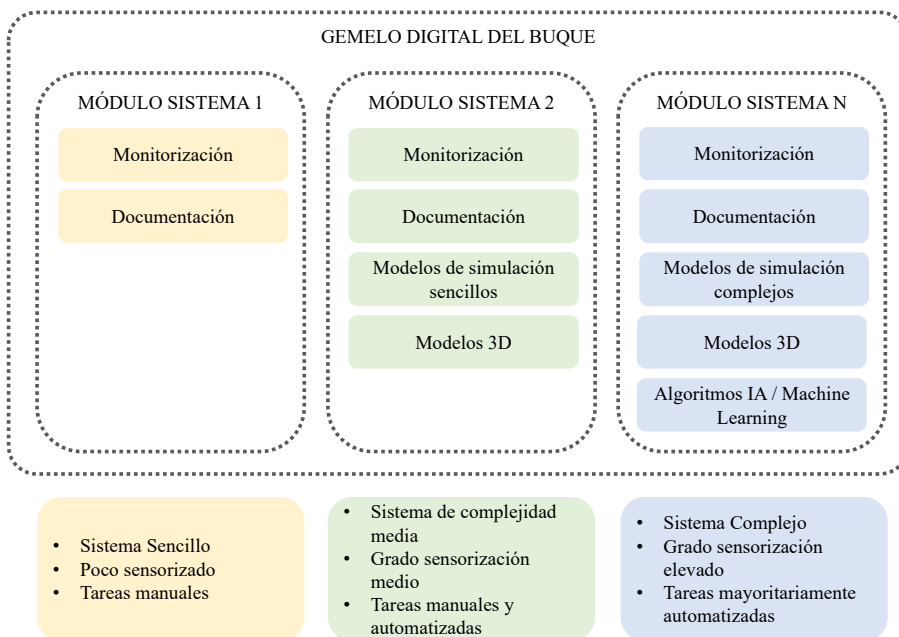
- La frecuencia de muestreo de señales no tiene por que ser la misma (y de hecho no debe de serlo) para todos los sistemas del buque. Deberá seleccionarse la frecuencia óptima para cada caso, de forma que no se empleen recursos de forma innecesaria.
- La heterogeneidad de sistemas y equipos hace que en el buque pueda haber sistemas altamente sensorizados, con características 4.0 o sistemas inteligentes, pero que también existan sistemas en el buque predominantemente manuales y con poca o nula sensorización.

Alcanzar el mayor grado de integración de los diferentes sistemas del buque es uno de los principales objetivos en el desarrollo del gemelo digital del buque y del desarrollo del concepto de buque inteligente.

### **3.6.3. Gemelo Digital**

Aunque la estructura de los componentes del gemelo digital de un buque seguirá el esquema general presentado en la imagen 3-8, dependiendo del sistema que se esté considerando, los correspondientes gemelos podrán ser más o menos complejos y contarán con más o menos funcionalidades. Como se adelantó al principio de este apartado correspondiente al gemelo digital del buque, se puede decir que este será un agregado de gemelos digitales de sistemas, pero hay que tener en cuenta que alguno de estos gemelos de sistemas (en el caso de sistemas no críticos, poco automatizados) podría estar formado únicamente en una monitorización del sistema y, por ejemplo, integración de la documentación de ese sistema.

En el caso de sistemas vitales del buque (por no entrar en el complejo mundo de los sistemas de combate de los buques militares) y con elevado grado de control y automatización, se puede tener un gemelo digital ligado a él que integre, además de lo mostrado en el ejemplo más sencillo (la monitorización y documentación anterior): visualización de modelos 3D, modelos de simulación del sistema, algoritmos basados en IA para, por ejemplo, detección o predicción de fallos, etc. En el siguiente esquema (imagen 3-10) se muestra un ejemplo:



**Imagen 3-10: Ejemplo de gemelo digital para varios sistemas del buque, con diferentes funcionalidades**

En los 3 ejemplos que se muestran en la imagen 3-10 se tratan de representar 3 posibles gemelos digitales de sistemas en el buque. En este caso, dichos ejemplos se han planteado siguiendo los criterios que se muestran en la parte baja de la figura y que responden a lo siguiente:

- Grado de complejidad del Sistema.
- Grado de sensorización.
- Grado de automatización del sistema.

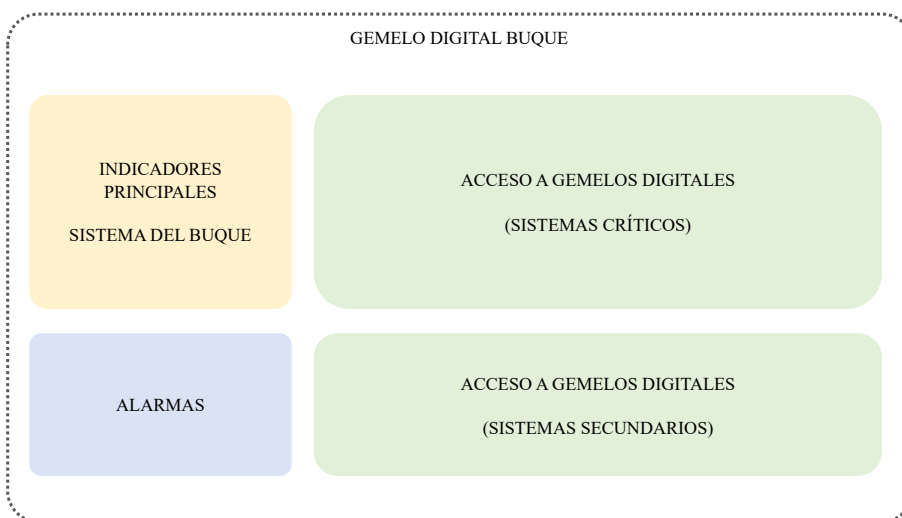
En general, aquellos sistemas más sencillos, que no cuentan con un elevado grado de sensorización y/o que las tareas que desempeñan no tienen un elevado grado de automatización, no permitirán desarrollar gemelos digitales plenamente funcionales. Se propone en este ejemplo, para esta tipología de sistemas en el buque, un gemelo digital basado únicamente en monitorización de alguna señal y documentación relativa a funcionamiento, mantenimiento, etc.

Sistemas con un grado de complejidad medio, representan sistemas que en general disponen de una sensorización limitada, pero que sin embargo puede ser suficiente para que los modelos de simulación desarrollados para su gemelo digital permitan realizar comparativa entre las señales reales y los resultados de esos modelos de simulación sencillos. No se consideran en este grupo de sistemas que requieran de una elevada frecuencia de muestreo de señales, o que sean sistemas críticos dentro del buque (en los cuales se debería de disponer de un gemelo digital de superiores prestaciones).

Para los sistemas críticos del buque, se optaría por desarrollar un gemelo digital como el que se muestra en el 3º caso. Se tratará de sistemas altamente automatizados, con elevado número de sensores (y con altas frecuencias de muestreo) que permitan la recogida de datos de funcionamiento en tiempo real, de modo que estos datos sean válidos para ser tratados por algoritmos de aprendizaje, ya sea para detección de anomalías, para predicción de comportamiento o incluso, para mantenimiento predictivo. Los gemelos digitales de este tipo de sistema podrán contar con modelos de comportamiento de detalle (debido al elevado conocimiento del sistema, así como a la disponibilidad de datos de funcionamiento que permitan refinar el modelo). Además, en general, se dispondrá de modelos 3D del sistema.

#### **3.6.3.1. Integración de funcionalidades. Interfaz de usuario**

La forma de acceso a la información dentro del gemelo digital del buque debe partir de un interfaz común. En esta interfaz inicial parece lógico presentar aquellos indicadores que se consideren los más importantes para representar el estado general del buque, además de contar con un acceso a los diferentes sistemas del buque. Dependiendo de la tipología de buque, podrá ser interesante agrupar estos sistemas de diferentes formas, aunque dado que no se considera un aspecto crítico, no se abordará en este trabajo. Desde esa interfaz inicial será posible acceder a los DTI de cada sistema.

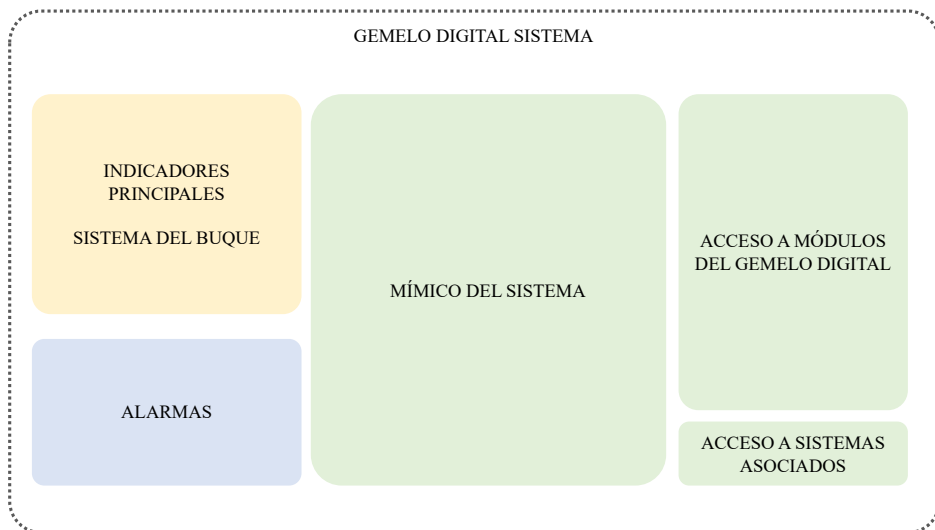


**Imagen 3-11: Diseño para interfaz gemelo digital del buque - pantalla principal**

En la figura 3-11 se presenta de la forma más simplificada posible los componentes con los que debería contar (como mínimo) la interfaz principal del gemelo digital del buque, y que deben incluir:

- Una visualización que presente el estado actual del buque, así como aquellos posibles avisos de relevancia (por ejemplo: mantenimientos pendientes) y alertas que pueda haber activas en ese momento.
- Un acceso a los diferentes sistemas del buque. A modo de ejemplo, en la imagen 3-11 se separa el acceso a los sistemas críticos y a los no críticos, aunque será la tipología de buque la que determine cómo se agrupan los sistemas en el entorno del gemelo digital.

Dentro de cada sistema se accederá a una nueva interfaz (como la que se puede ver, de nuevo, a modo de ejemplo, en la imagen 3-12) en la que se presenten:



**Imagen 3-12: Diseño para interfaz gemelo digital del buque – pantalla gemelo digital de sistema**

- Señales de estado del sistema concreto.
- Mímico del Sistema con monitorización.
- Accesos a los demás componentes (modelos de simulación, algoritmos de aprendizaje, modelos 3D, etc.).

### 3.6.3.2. Modelos de Simulación

En general, en el caso de los sistemas del buque, estos serán sistemas complejos multifísicos, es decir, que se componen de múltiples modelos físicos (eléctrico, mecánico, térmico, hidráulico, etc.). En función del sistema que se esté considerando puede ser interesante (o necesario) emplear una u otra herramienta de simulación.

Pero será a la hora de plantear la integración de modelos de simulación en el entorno del gemelo digital del buque, cuando se presenten los principales retos. Dado que el buque es un sistema complejo, puede que la simulación de uno u otro sistema se haya llevado a cabo con una herramienta diferente. No se trata de un problema específico de la simulación de sistemas del buque, pero sí es un claro ejemplo de lo que a menudo sucede en sistemas complejos. Será inevitable encontrarse en muchos casos con múltiples plataformas heterogéneas, que a menudo se emplean para simular sistemas que incluso pueden llegar a interactuar. Pero es difícil, o casi anecdótico, que se comuniquen

entre ellas. En las fases de diseño no tiene porque ser un problema, ya que estos se pueden dimensionar empleando datos provenientes de modelos de los sistemas con los que interaccionan sin que se necesite que lleguen en tiempo real. Pero los problemas surgen, cuando se quiere que ambos sistemas se puedan simular de forma simultánea en tiempo real, o incluso que se quieran emplear para predecir un comportamiento a lo largo del tiempo. En estos casos será imprescindible que ambos modelos se comuniquen entre ellos.

Además, aunque resulte obvio, a la hora de pensar en la implantación del gemelo digital del buque, a bordo, no es una posibilidad tener que disponer a bordo de las diferentes herramientas que se hayan empleado para simular cada uno de los sistemas del buque.

Estos dos retos pueden ser abordados empleando un estándar como es el FMI, mediante el cual se pueden exportar diferentes modelos de simulación procedentes de diferentes plataformas, importarlos o llevar a cabo co-simulación. Schweiger llevó a cabo un análisis de las principales necesidades de investigación relacionadas con la co-simulación de FMUs y que han sido identificados por los actores implicados (Schweiger et al., 2019). No se trata de un estudio específico del sector naval, pero si de la co-simulación de sistemas complejos, por lo que se considera apropiado tenerlo como referencia. Se resumen a continuación las principales conclusiones de este análisis en el que se destaca que los expertos consultados consideran el estándar FMI como el estándar más prometedor tanto para simulaciones en tiempo continuo, eventos discretos y co-simulación:

- Se identificó como un problema habitual las dificultades para abordar aspectos prácticos.
- Las necesidades de investigación más importantes identificadas han sido: la comprensión teórica en diferentes enfoques de co-simulación y validación de modelos.
- El aspecto identificado como principal dificultad está relacionado con aspectos prácticos, mientras que la necesidad de investigación más reclamada se relaciona con la comprensión teórica.
- Los resultados del análisis determinaron que predominan los aspectos que definen las fortalezas y oportunidades. Los expertos asignan lo



más importante a la necesidad de herramientas fáciles de usar que puedan incluir, principalmente, algoritmos maestros predefinidos.

Por todo lo comentado, es evidente que la parte de integración de los modelos de simulación en el entorno del gemelo digital cobra una especial relevancia en el caso del buque. En este campo, existe un importante desarrollo, ya citado anteriormente, la Open Simulation Platform.

### 3.6.3.3. Almacenamiento de datos

El almacenamiento de datos será un punto de gran importancia dentro del desarrollo de gemelos digitales en buques, debido a que un buque navegando no se encuentra en las mismas condiciones que un sistema en tierra a la hora de almacenar la información generada. Antes de iniciar una navegación será necesario garantizar que se contará con capacidad suficiente para el almacenamiento de los datos que se van a generar durante el tiempo que el buque se encuentre navegando.

En la siguiente tabla se ha estimado de forma aproximada los requisitos de almacenamiento de un gemelo digital del buque, para un buque al que se le ha supuesto una navegación de una duración de 6 meses.

**Tabla 3-1. Parámetros del gemelo digital (estimación) (Trueba-Martinez et al., 2020)**

<b>Parámetro</b>	<b>Valor aproximado</b>
<b>Tipo de base de datos</b>	De serie de tiempo
<b>Número de señales</b>	40000
<b>Tamaño por señal</b>	4 bytes (7 dígitos precisión)
<b>Frecuencia</b>	20 Hz
<b>Almacenamiento en seis meses</b>	48 TB

En la tabla se presenta un valor aproximado de los requisitos de almacenamiento, suponiendo que el buque no realice un volcado de datos durante seis meses, y se almacenen todos los datos de los sensores “en bruto” con la misma frecuencia. No se ha tenido en cuenta que posiblemente no todos los sensores tengan la misma frecuencia o la posibilidad de almacenar sólo los

datos de los sensores de sistemas críticos del buque. La capacidad de procesamiento que requieren los modelos de simulación o la aplicación soporte del gemelo no es un requisito crítico.

#### **3.6.3.4. Análisis de Datos y Algoritmos de aprendizaje y predicción**

En aquellos sistemas con capacidades de IoT y un número de sensores adecuado, será posible emplear métodos de análisis de datos y de IA para el análisis del funcionamiento real del sistema. Aunque no será posible en todos los sistemas del buque, debido que las necesidades para la implementación de técnicas de analítica de datos (o aún más, en el caso de big data) pero también para la aplicación de diferentes algoritmos de inteligencia artificial (como puede ser el caso de machine learning) pasan por dotar al sistema de una elevada sensorica y, además de ello, en ocasiones, que esta sensorica tenga una frecuencia de muestreo adecuada para el análisis que se desee realizar de los datos (en el caso de querer analizar, por ejemplo, patrones de funcionamiento).

Pero en general, se valorará la aplicación de métodos de analítica de datos que permitan detectar desviaciones respecto a la información recogida de funcionamiento normal. Se podrán establecer reglas que permitan identificar problemas asociados a determinados síntomas o que permitan llevar a cabo predicción de fallos.

Sin embargo, la fortaleza que pueden presentar los algoritmos basados en inteligencia artificial para la predicción de fallos y para el control de la operación del buque, se basa, en que no es necesario llevar a cabo una tarea compleja y a las veces difícil de definición de reglas (que suelen pasar por un exhaustivo análisis casi manual de los datos), sino que se pueden etiquetar datos de forma masiva, o incluso emplear algoritmos de inteligencia artificial no supervisados, los cuales son capaces de aprender sin necesidad de ser entrenados de forma supervisada.

#### **3.6.3.5. Conexión con Sistemas de Información y Logística**

Aunque en otros sectores puede no ser especialmente relevante, en el caso de la aplicación de gemelo digital al buque, cobran gran importancia los aspectos relacionados con la conexión entre el gemelo y los sistemas de información existentes.

Esta conexión debe existir desde las primeras fases de diseño del buque y de sus sistemas. En el momento en el que exista el proyecto en el gestor del ciclo de vida, PLM, debería realizarse la conexión gemelo/PLM, con lo cual la información correspondiente al sistema físico aún en fase de desarrollo. En la fase de construcción del buque, a través de la toma de datos de sensores del buque, se puede ir supervisando el proceso y comparando con el diseño inicial.

Por otro lado, lograr la conexión del gemelo digital con el ERP de la empresa, permitirá mantener operativo el gemelo digital durante la fase de construcción del buque, pudiendo ver el avance de los diferentes componentes y sistemas del buque hasta el grado de detalle deseado (órdenes de trabajo, KPIs, etc.).

Alcanzar estos 2 objetivos permitirá poder hacer referencia al buque en diseño/fabricación/operación como buque conectado.

#### **3.6.4. Otros aspectos relevantes en la implantación del GD del buque. ¿Gemelo digital del buque embarcado o en tierra?**

Un aspecto importante en la implantación del gemelo digital del buque estará relacionado con la localización geográfica en la que se encontrarán los diferentes componentes que forman el gemelo. Será necesario conocer si hay o si es necesario que haya una localización en tierra en la que exista una réplica del gemelo digital y, si es así, será necesario analizar qué componentes se encontrarán en cada localización, cuales deberán de estar duplicados (compartiendo o no las mismas funcionalidades) y cómo se llevará a cabo el traspaso de información entre ambos. Es evidente que, en todo caso, dependerá principalmente de la tipología de buque y del interés que el armador, naviera, astillero, armada, etc. requieran del gemelo digital.

A priori, existen componentes de la arquitectura del gemelo digital, que deberán, en todos los casos, estar desplegados y ejecutados a bordo. En primer lugar, los sensores, PLC y sistemas de comunicación hacia el gemelo, evidentemente estarán a bordo. Los datos recopilados mediante esos sensores deberán ser procesados por los modelos numéricos del gemelo digital (ya sean modelos de simulación de comportamiento, o modelos de análisis de datos); teniendo en cuenta que el operario del gemelo puede requerir de la realización

de simulaciones que permitan, por ejemplo, el análisis de anomalías, parece claro que estas capacidades del gemelo también deberían estar a bordo.

No está tan clara la ubicación de los algoritmos de predicción, de ajuste de modelos del gemelo y de análisis de mantenimiento. Dado que estos elementos pueden no demandarse en el muy corto plazo, las herramientas de ajuste, procesado de datos, análisis de mantenimiento y similares, podrían situarse en una base en tierra, recibir los datos del buque cuando fuese posible, y devolver los resultados al mismo cuando estuviesen disponibles.

En lo referente a la localización del almacenamiento de los datos; será necesario determinar si se desea almacenar datos brutos, tanto de sensores reales como de simulaciones del gemelo que se realicen a bordo, a la frecuencia de muestreo o de generación, o si se desea almacenar únicamente estadísticos de los mismos. Estas decisiones estarán condicionadas en gran medida por la decisión tomada para la localización de los modelos. Aquellos datos que se requieran en tierra para su análisis deberán ser almacenados en el buque, y será necesario determinar si esos datos podrán almacenarse a bordo en su totalidad hasta la llegada a puerto en la que se realice el volcado de datos, o si se puede requerir el envío a tierra una cantidad de ellos, ya sean datos brutos o los ya mencionados estadísticos. Estas decisiones condicionarán las necesidades de conectividad con la base de tierra y las necesidades de espacio de almacenamiento.

Una de las principales ventajas que permite el despliegue de un gemelo digital en tierra será la conexión total con el PLM para la logística de mantenimiento o con el ERP para la localización de repuestos en stock. En este caso se entiende que no hay restricciones de almacenamiento, como en el caso del buque, pero sí se debería utilizar el mismo tipo de base de datos en ambos casos para garantizar la accesibilidad a la información.

Finalmente, por citar los componentes del gemelo digital definidos, los sistemas de visualización e interfaz de usuario también deberían estar disponibles en el buque y el gemelo en tierra, considerando que los roles de usuario que empleen uno u otro sistema podrán ser diferentes, por lo que dicha interfaz deberá ser diseñada para los requisitos concretos del perfil de usuario.

En la tabla 3-2 se resumen los componentes del gemelo digital, así como su posible ubicación en el caso de contar con un gemelo digital embarcado y otro en tierra.

**Tabla 3-2: Resumen de ubicación de componentes del gemelo digital**

Componente de la arquitectura	A bordo	En tierra
Adquisición de datos y comunicación con el sistema físico	x	
Modelos de simulación y/o algoritmos de procesado	x	x
Almacenamiento de datos	x	x
Conexión con PLM y/o ERP		x
Interfaz	x	x

### 3.7. PLATAFORMAS DE IMPLEMENTACIÓN DEL GEMELO DIGITAL

El despliegue de un gemelo digital requerirá la integración de los diferentes componentes que lo forman. Es evidente que será necesario emplear una plataforma de integración que permita:

- La conexión con el sistema real.
- La conexión con los motores de simulación de modelos.
- El acceso al lugar donde se almacene la información (históricos, modelos 3D, documentación, etc.).
- La visualización e interfaz con el usuario de todo lo anterior.

Las plataformas que se suelen emplear en desarrollos de gemelo digital son las mismas que en el caso de despliegues de IoT, que suelen contar con:

- Conexión con gateway que permite el enlace con el sistema de adquisición de datos, es decir, la puerta de entrada de los datos del sistema físico a la plataforma de integración.
- Elementos para visualización de modelos 3D.
- A menudo permiten la conexión con el resto de plataformas, PLM y ERP.
- Conexión con motores de simulación de modelos.

Las plataformas de implementación podrán ser, en general, de dos tipos: por una parte, plataformas comerciales de IoT y por otra parte una solución de desarrollo propio. Las plataformas comerciales de IoT proveen de algunas de las funcionalidades presentadas en la arquitectura con la contrapartida del coste de licencia por este software. Las soluciones propias no tienen un coste de licencia ni de mantenimiento, pero sí un coste de implementación inicial, aunque se pueden personalizar más a las funcionalidades que se requieran.

## 4. IMPLEMENTACIÓN DEL GEMELO DIGITAL

---

### 4.1. DESCRIPCIÓN DEL CASO DE USO

El objetivo principal de este primer caso de uso busca comprobar la aplicación práctica de la metodología descrita en el capítulo 3. Para ello, se evaluará la implantación de un gemelo digital de un sistema reducido de los múltiples presentes en un buque, y que en este caso consistirá en un sistema de HVAC (aire acondicionado: heating, ventilation and air conditioning). Sobre este sistema se han ido desplegando diferentes escenarios para analizar cómo podría ser la implantación de un gemelo digital en un buque. Este gemelo digital sería uno de los agregados que se describen en la imagen 3-9, y que, según lo indicado en la imagen 3-10 representa un sistema de media complejidad.

Durante la primera fase del desarrollo, los trabajos se centraron en el desarrollo de modelos de simulación del sistema real, así como de la adquisición de datos desde el sistema físico hacia el gemelo digital (ubicado en una plataforma de IoT comercial). En esa primera fase, la plataforma seleccionada para la prueba de concepto fue Thingworx de PTC y los modelos de simulación se desarrollaron en OpenModelica.

En una segunda fase, se pasó de un despliegue centralizado en una plataforma de IoT a una propuesta descentralizada. En este caso, el trabajo estuvo centrado sobre todo en el desarrollo de modelos de simulación y en su ejecución en el entorno del gemelo digital.

Este capítulo se estructurará de forma similar a lo planteado en la arquitectura de gemelo digital del capítulo 3. En cada uno de los principales componentes de la arquitectura propuesta se describirán el planteamiento o las diferentes alternativas planteadas que se consideraron en el desarrollo del caso de uso relacionado con el sistema HVAC.

## 4.2. SISTEMA FÍSICO

Se propone la aplicación de la metodología desarrollada a un sistema para aire acondicionado con un circuito cerrado de agua refrigerada. El objetivo de un sistema de este tipo es ofrecer al buque un sistema de calefacción y climatización. Su funcionamiento es doble: podrá funcionar como equipo de calefacción o como equipo de aire acondicionado. La elección de este sistema como caso de uso presenta las siguientes ventajas:

- Se trata de un sistema que abarca elementos mecánicos, hidráulicos, eléctricos y térmicos. Esto es especialmente interesante en lo que corresponde con los modelos de simulación del sistema, y que permitirá probar alguna de las herramientas de simulación de sistemas multifísicos de las que se ha hablado en el capítulo 3.
- Relacionado con el punto anterior, y dado que se trata de un sistema en el que los componentes están claramente diferenciados y podrían ser adquiridos a diferentes suministradores, parece un claro candidato para aprobar la co-simulación de modelos de sistemas (de la que también se ha hablado en el capítulo 3) y con ello simular que, en el caso de un sistema real, se están importando diferentes modelos desarrollados por el fabricante de cada uno de los componentes los cuales deben de ser orquestados correctamente para llevar a cabo la co-simulación.

Es importante señalar que este sistema incorpora un elevado nivel de control sobre el sistema, superior al que suelen contar este tipo de sistemas cuando se implementan en el buque. La finalidad de esto es, además de optimizar el funcionamiento del sistema, validar modelos de simulación que puedan incorporar este control.

### 4.2.1. Descripción del sistema

El sistema fue diseñado y fabricado por Navantia e instalado en el Centro de Innovación y Servicios de A Cabana (Ferrol) con la idea de servir como punto de partida para el desarrollo de trabajos relacionados con las tecnologías que engloban el gemelo digital. En la imagen 4-1, se muestra un esquema de la instalación, se trata de una cabina la cual contiene en su interior un sistema HVAC que sirve para enfriar o calentar el aire que se encuentra en el interior de dicha cabina, haciéndolo pasar en el unit cooler a través de una resistencia



eléctrica o de un serpentín de tubos y aletas, según el modo en el que esté trabajando el sistema (modo verano o modo invierno).

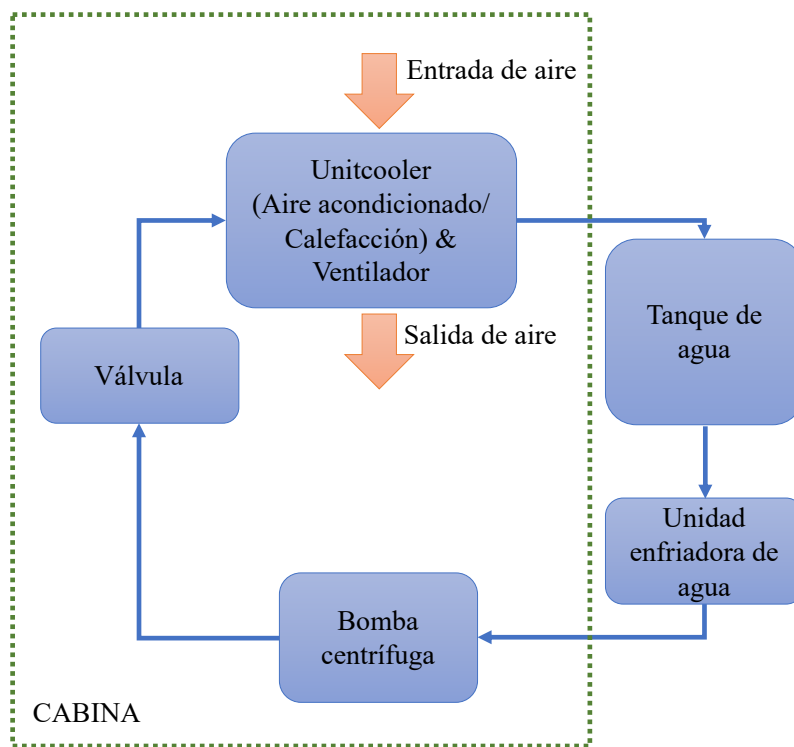


Imagen 4-1: Esquema del sistema HVAC

En general, los componentes de este sistema en el caso de un buque serán los siguientes:

- **Unit cooler (UC).** Se trata de un elemento formado por un intercambiador de calor de aletas y tubos (aire/agua), una resistencia eléctrica y un ventilador centrífugo. Este equipo incorpora capacidades 4.0 adicionales, por ejemplo, la capacidad de demandar un caudal determinado de agua en función de la relación entre la temperatura de la cabina y la temperatura consigna. Para ello, este sistema cuenta con un PLC propio que le permite lo siguiente:
  - Monitorización de señales de los sensores: temperaturas del agua (en tuberías de entrada y salida), caída de presión en la batería (en tuberías de entrada y salida), temperaturas del aire (entrada y salida), monitor de vibración del ventilador,

- monitor de vibración del motor, RPM del ventilador, presión del ventilador (entrada / salida).
- Recepción de datos externos, desde el PLC del arrancador: caudal de agua medido por la válvula, corriente de la resistencia eléctrica (heater), revoluciones del motor, temperatura y humedad de la cabina.
- **Bomba centrífuga** para la circulación de agua por el sistema. La bomba incorpora transmisores de presión para succión y descarga. Los transmisores envían al arrancador las siguientes señales: presión de succión, presión de descarga y temperatura de los devanados del motor.
  - **Válvula motorizada** para modulación del flujo de agua. Cuenta con las siguientes capacidades:
    - Monitorización de señales: monitorización estándar del actuador (porcentaje de apertura) y medición de la caída de presión en la válvula. En función de la caída de presión la válvula calcula el caudal de agua correspondiente.
    - Recepción de datos externos (desde el PLC de arranque): comando de demanda de flujo del unit cooler.
  - **Arrancador.** Cuenta con lo siguiente: módulo de potencia, módulo de control de velocidad del ventilador, PLC con automatización local y conexión de sensores, pantalla táctil (HMI) para monitorización y control local, y como sensores los siguientes: sensores de voltaje, intensidad y de temperatura del armario. El arrancador incorpora además el módulo de comunicaciones para enlace con el gemelo digital. El arrancador cuenta con las siguientes capacidades:
    - Monitorización de señales de sensores: consumo eléctrico de motores y calentador, temperaturas de bobinado de motores, alarma de nivel bajo del tanque de expansión, presiones de succión y descarga de la bomba.
    - Recepción de datos externos: caudal de agua medido por la válvula, demanda de rpm del ventilador, demanda de flujo del unit cooler, demanda de arranque / parada de la bomba.
  - Tuberías **de plástico**, estándar comercial.
  - Otras instalaciones de soporte:

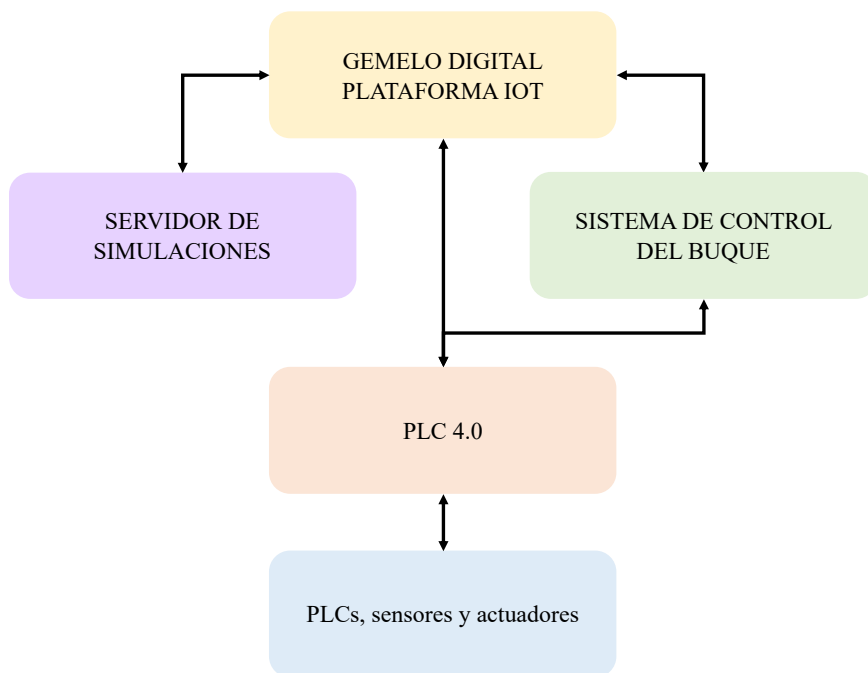
- **Cabina** con aislamiento térmico con: puerta de acceso, ventana de observación, suministro de aire fresco del exterior, ventilación de escape natural, iluminación, medición de temperatura, medición de humedad.
- **Tanque de inercia** para el agua del sistema y **unidad de refrigeración externa**, que permite enfriar el agua que pasa por el unit cooler.
- Unidad de control local que realiza funciones al mismo nivel que el **sistema de control del buque** a bordo.
- **Alimentación eléctrica**

### 4.3. DESCRIPCIÓN DE LAS ARQUITECTURAS PLANTEADAS PARA EL DESPLIEGUE DEL GEMELO DIGITAL

Se analizaron dos alternativas para la arquitectura que compone el gemelo digital de este caso de uso, una primera arquitectura en la que el gemelo digital está centralizado en una plataforma de IoT y una segunda alternativa con una arquitectura no basada en plataforma IoT (descentralizada).

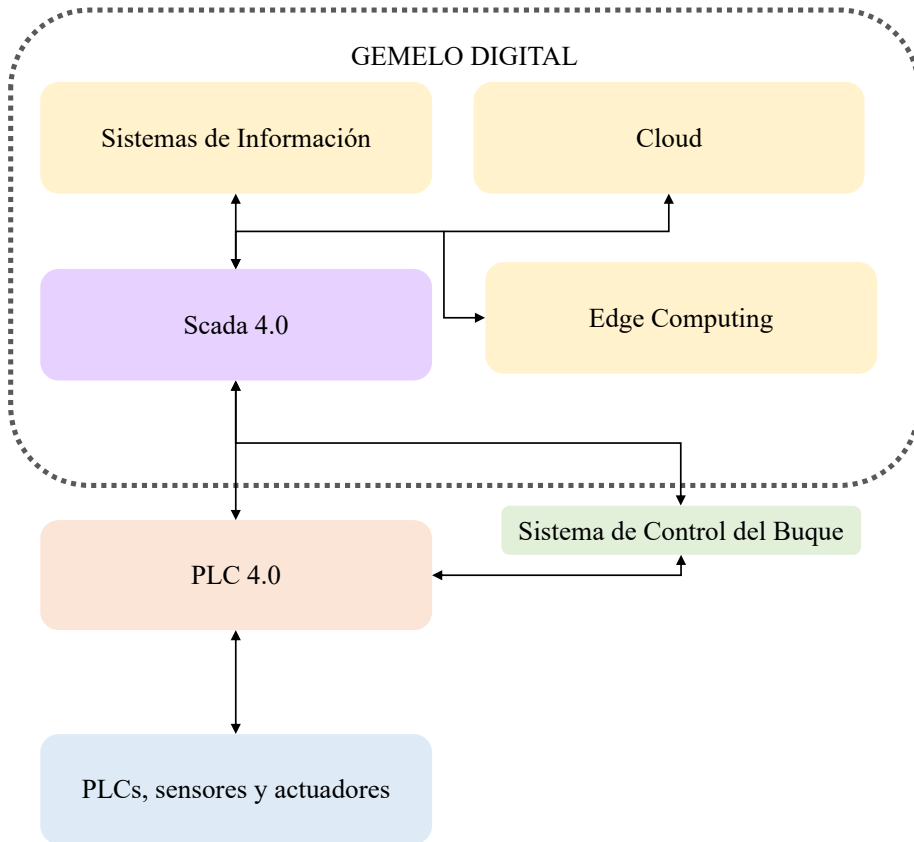
En el primer planteamiento realizado, basado en el uso de la plataforma comercial de IoT de PTC, Thingworx, el esquema general que define la arquitectura necesaria para el despliegue del gemelo digital se corresponde el que se resume en la imagen 4-2.

En este caso, el gemelo digital se enmarca en su totalidad dentro de la plataforma IoT, sobre la cual se desarrollan o implementan las diferentes funcionalidades: adquisición de datos, almacenamiento, ejecución de modelos de simulación, conexión con sistemas de información, etc. Siendo siempre la plataforma de IoT, además, el interfaz desde el que se accede a todos los componentes. Además, en la arquitectura planteada, se contó con lo que se ha denominado "PLC 4.0", un PLC de última generación, que integra características 4.0 y que sirve como punto de unión entre la capa física y la digital. Debido a que este desarrollo piloto estaba orientado a la aplicación del gemelo digital en los buques fabricados por Navantia, se contempló que el flujo desde/hacia el gemelo digital pudiera pasar por el Sistema de Control de Plataforma del buque.



**Imagen 4-2: 1ª arquitectura planteada: gemelo digital centralizado en plataforma de IoT**

Por otro lado, en el segundo planteamiento realizado en este sistema, la arquitectura planteada se modificó de la siguiente forma: el núcleo del gemelo digital ya no se encuentra en una plataforma de IoT, sino que se encuentra descentralizado. Sin embargo, sigue existiendo un nexo de unión para los componentes que integran el gemelo digital (donde se centralizan los flujos de información y la visualización de los datos del gemelo), pero en este caso ese nexo se encuentra en el Sistema SCADA de Siemens. El planteamiento que propuso Siemens para el gemelo digital plantea que el gemelo digital se encuentre en una capa baja de la arquitectura, se añade por lo tanto a la infraestructura del gemelo digital una capa de computación edge, en la que se centraliza parte del contenido del gemelo digital que en el planteamiento inicial se encontraba en la plataforma IOT. En el siguiente esquema (imagen 4-3) se resume la arquitectura planteada por Siemens:



**Imagen 4-3: 2ª arquitectura planteada: gemelo digital descentralizado**

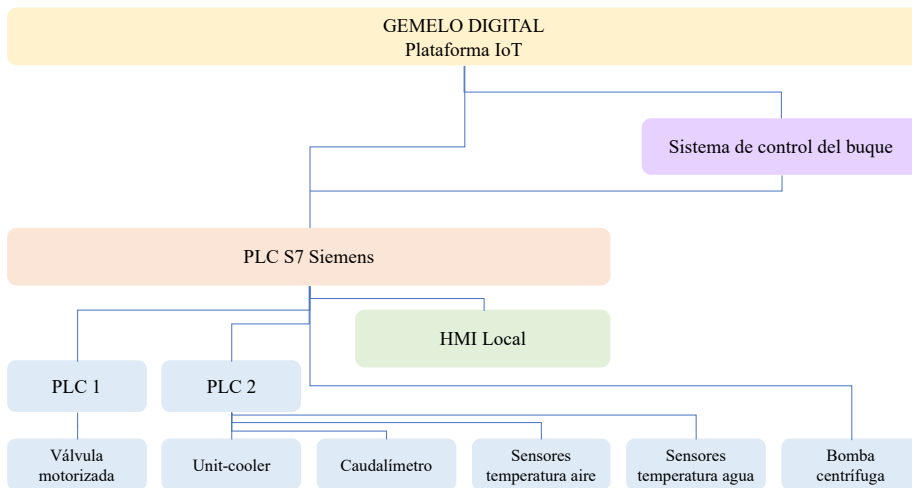
Los trabajos realizados en esta tesis, como se ha adelantado, ya no se centraron en validar desarrollos para las diferentes partes de la arquitectura, si no que estarán centrados en el desarrollo de modelos de simulación para ser empleados en el gemelo digital y en cómo se pueden ejecutar después esos modelos en este entorno (edge computing).

#### 4.4. DESPLIEGUE CENTRALIZADO EN PLATAFORMA IOT

El planteamiento a la hora de establecer la arquitectura de un gemelo digital no es ni mucho menos único, aunque es cierto que a menudo se proponen diseños basados en una plataforma de IoT (en general esa idea después se complementa con desarrollos en la capa local-edge, sobre todo cuando se llevan a cabo despliegues en planta). En las siguientes referencias se puede encontrar información sobre plataformas IoT para desarrollo de gemelos digitales: Mindsphere de Siemens (Siemens, 2018, 2020c), Thingworx de PTC

(Dertien et al., 2019; Immerman & Lang, 2019), Predix de General Electric (General Electric, 2018; General Electric Company, 2015), IoT Watson de IBM (Bleakley, 2020; IBM, 2020), o la plataforma libre DITTO de Eclipse (Jaeckle, 2020).

La plataforma IoT seleccionada para arrancar el desarrollo del gemelo digital del buque fue Thingworx. Se optó por que esta fuera el núcleo del gemelo digital, de forma que los diferentes componentes se desarrollaran o fueran accesibles desde la plataforma de IoT. Por lo tanto, fue en esta plataforma donde se centralizó la adquisición, almacenamiento y visualización de datos, y desde donde se realizó la conexión con los modelos de simulación, con los modelos 3D, o con otros sistemas de información empleados de los que se puede requerir información (PLM, ERP, etc.). La arquitectura detallada de esta aproximación es la que se presenta en la imagen 4-4:



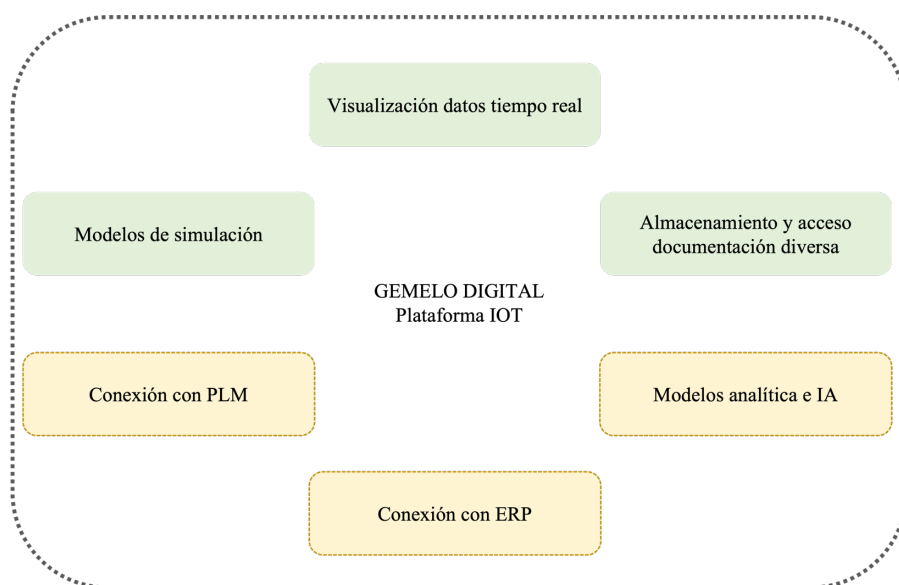
**Imagen 4-4: Arquitectura despliegue gemelo digital (gemelo digital centralizado en plataforma IoT)**

En el esquema correspondiente a la imagen 4-5 se puede observar la idea inicial que se planteó en lo referente a la arquitectura o funcionalidades con las que debería de contar un desarrollo de este tipo y que se enumeran a continuación:

- Monitorización del sistema real.
- Visualización de modelos 3D de los componentes.

- Acceso y visualización de documentación del sistema: especificaciones, manuales de funcionamiento/mantenimiento, etc.
- Modelos de simulación del Sistema.
- Modelos de análisis de datos y/o modelos basados en IA.

El trabajo desarrollado se centró en conectar los datos de los sensores y actuadores del sistema real con la plataforma y, sobre todo, en el desarrollo de modelos de simulación que permitieran, por un lado, comparar los datos reales obtenidos con los correspondientes a las simulaciones, y por otro, predecir el comportamiento del sistema.



**Imagen 4-5: Componentes/funcionalidades planteados en el gemelo digital**

#### 4.4.1. Modelo de Datos y desarrollo de diagramas conceptuales

Según la clasificación elaborada por Adamenko, a la hora de plantear un gemelo digital se suele hablar de gemelos basados en datos y gemelos basados en sistemas (Adamenko et al., 2020). En el caso de un gemelo digital basado en datos, el punto de partida del desarrollo son una serie de datos procedentes de los sensores de los que se dispone, y que se estructuran siguiendo un cierto criterio, como puede ser, agrupar los datos por funcionalidades o ensamblajes del sistema físico, generando así el modelo de datos correspondiente. Por otro lado, un gemelo digital basado en sistemas, el foco se posiciona en el sistema físico, de este modo se componen modelos en torno a él. Para desarrollar

modelos de este tipo es necesario conocer el sistema real, sus conexiones, funcionamiento, control, etc. Este tipo de modelos se ajusta a lo que se busca en este caso de uso y, como se verá más adelante, plantear el gemelo digital de esta forma presentará importantes ventajas a la hora de desarrollar los modelos de simulación del sistema.

Siguiendo este razonamiento, una de las principales tareas que se desarrolló fue plantear una estructura de datos, con las diferentes señales de las que se dispone, agrupadas por sistemas. Para el desarrollo de este modelo de datos del sistema HVAC se hizo uso de una herramienta MBSE (model based system engineering o diseño de sistemas basado en modelos) (Madni et al., 2019; Y. Wang et al., 2021). En concreto, se empleó para este caso la herramienta Modelio (Modeliosoft Consulting Team, 2009). Este tipo de herramientas, basadas en el lenguaje SysML (Delligatti, 2014) permiten desarrollar diferentes diagramas: diagramas de bloques, diagramas de estado, diagramas de actividad y otra tipología de diagramas basados en MBSE que pueden resultar de utilidad en el desarrollo de proyectos en los que se emplea esta metodología o, en general, en cualquier tipo de desarrollo basado en sistemas, como por ejemplo, en el diseño de sistemas complejos (Bilic et al., 2019).

En la imagen 4-6 se puede ver el esquema general de definición de variables para el modelo de datos realizado en Modelio.

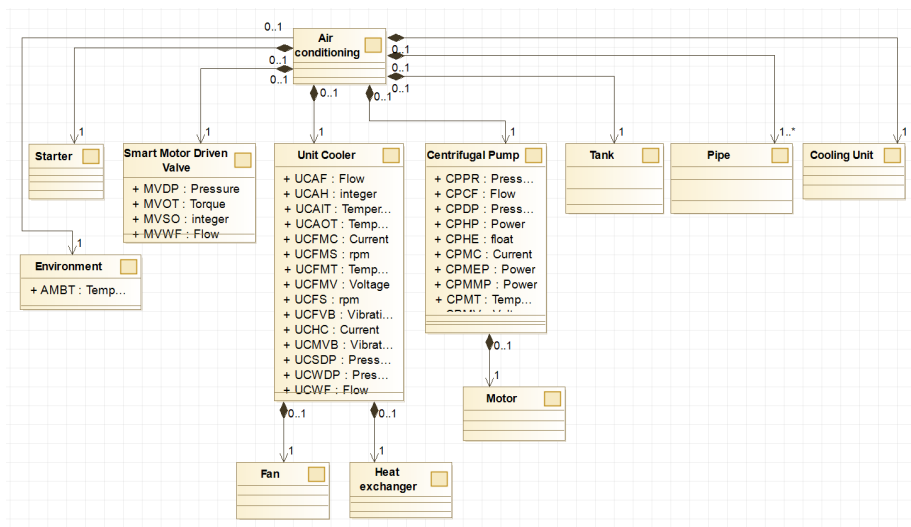


Imagen 4-6: Diagrama de variables para el modelo de datos (SysML)



Dentro del modelo de datos definido se encuentra toda la información al respecto de las diferentes variables empleadas en el sistema: nombre de la variable, componente del que depende, señal sensorizada o calculada, etc. En el apéndice A se puede consultar una tabla con esta información.

#### 4.4.2. Adquisición de datos de campo

Como se ha adelantado al principio del capítulo, la capa física es común a ambos planteamientos, en ella se encuentran los diferentes componentes físicos y los sensores que se disponen en el sistema. Alguno de los componentes, en concreto, en el caso de la válvula y del unit cooler, estos cuentan con un PLC propio, que servirá como intermediario ante el PLC principal del sistema. Los PLCs se pueden considerar como un aparte importante del concepto industrial del IoT, especialmente cuando existen dispositivos que no poseen características inteligentes propias. Si una máquina no posee capacidades IoT o capacidades 4.0, el PLC puede aportársela. Esto ha sido posible por el avance que se ha producido en el campo de los PLCs en los últimos años. La capacidad de los PLCs ha aumentado de forma importante, igual que sus prestaciones, velocidad y capacidad de almacenamiento. (Azarmipour et al., 2019)

Los PLCs se encargarán de:

- Controlar los diferentes componentes de la capa inferior.
- Recopilar datos del estado de los componentes y de los sensores.
- Realizar cálculos básicos.
- Transferir la información a la capa superior.

En resumen, habrá 3 PLCs, uno que se encarga del control de la válvula y otro para el unit-cooler. Ambos se comunicarán directamente con el PLC del arrancador, el cual también recibe señales del sistema (por ejemplo, en el caso de la bomba, ya que esta no dispone de PLC propio). De este modo, habrá señales que vayan directamente al arrancador, y otras que pasen por estos PLCs. Esto se detallará en el apartado correspondiente al Modelo de Datos para adquisición.

El protocolo de comunicación empleado entre PLCs es MODBUS TCP/IP. El protocolo MODBUS es un protocolo de comunicación basado en una arquitectura maestro/esclavo o cliente/servidor. MODBUS TCP/IP es una variante de la familia MODBUS para un entorno 'Intranet' o 'Internet' utilizando los

protocolos TCP/IP. El uso más común es para la conexión Ethernet de PLC's (ABB, 2010; Thomas, 2008).

#### 4.4.3. Interfaz campo y gemelo digital

Para la conexión entre el sistema físico y el sistema digital, se consideraron dos posibilidades, ambas basadas en la conexión entre la capa física y la plataforma IoT que forma el núcleo del gemelo digital.

En la primera alternativa se estableció una conexión directa entre el PLC principal del sistema, en este caso un PLC 4.0 de fabricación Siemens (S7) con el PC en el que se encontraba la plataforma IoT para gemelo digital. Es decir, en este caso la adquisición de datos del gemelo digital se llevó a cabo directamente desde directamente sobre el sistema. Se planteó el uso del protocolo MQTT para esta comunicación.

En la segunda alternativa, se optó por comprobar cómo se realizaría la adquisición de datos hacia el gemelo digital, si en lugar de hacerla directamente desde el PLC 4.0 del sistema se hace pasar a través del sistema de control del buque. La comunicación entre el PLC y el sistema de control se realizó empleando el mismo protocolo MQTT, mientras que desde el sistema de control hacia el gemelo digital se abordó empleando el protocolo API-Rest (basado en http).

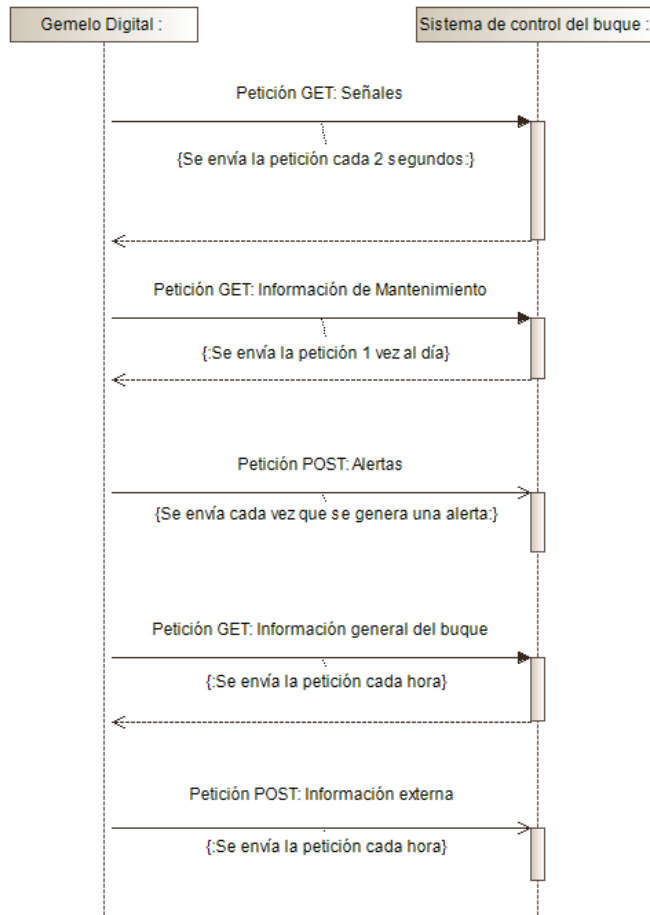
En concreto, en este desarrollo API-Rest se plantearon las siguientes reglas:

- 1) Petición de **información general del buque**. Se lanza una petición de tipo "GET" al sistema de control del buque, para que este devuelva información general, como puede ser velocidad, ruta, consumo de combustible... que puedan resultar de utilidad en el gemelo digital. En este caso concreto la temperatura exterior, podría ser utilizada como input del modelo de simulación.
- 2) Petición de información de **mantenimiento**. Petición tipo "GET" del gemelo digital al sistema de control del buque para que este devuelva información de operación acerca del sistema, agrupándose por máquina: mantenimientos programados, horas hasta próxima parada del sistema...
- 3) Información de **señales**: Petición del gemelo digital al sistema de control del buque para recibir las señales sensorizadas, agrupadas

por sistema. Esta petición ha de realizarse en un periodo corto (en esta propuesta inicial se tomó como referencia 2 segundos) para mantener actualizado el gemelo digital. De este modo, cada 2 segundos se lanza una petición de tipo “GET” y desde el sistema de control se devuelve un string con las señales y su valor actual (agrupadas por componentes según el modelo de datos).

- 4) Envío de **alertas**. Petición de tipo “POST” del gemelo digital al sistema de control para indicar una alerta. Las alertas contienen el nombre del parámetro que las ha provocado y su valor anómalo, así como el nombre del dispositivo en el que se han producido, una breve descripción y una asignación de severidad. Otros campos adicionales en el mensaje de la alerta son una identificación de la alerta y una indicación de tarea a realizar a causa de la alerta.
- 5) Envío de **información externa** al sistema de control. Petición de tipo “POST” del gemelo digital al sistema de control para proveer a éste de información externa que pueda darle contexto y poder optimizar su control o cambiar el modo de funcionamiento. Parámetros relevantes: día de la semana, estación del año, número de personas en el compartimento.

Estas reglas se implementan en Thingworx tal y como se plantea el diagrama de secuencias de la imagen 4-7. En el caso concreto de la petición de las señales del sistema real el proceso se resume en lo siguiente: Thingworx hace una petición de tipo GET cada 2 segundos con un temporizador al sistema de control y este devuelve un fichero JSON a Thingworx. Una vez que se recibe el JSON se almacena cada variable en su modelo de datos correspondiente (en Thingworx), pasando de la cadena de texto que devuelve el JSON a propiedades de tipo numérico (procesadas por Thingworx) y que de este modo se almacenan en el gemelo digital, de forma que se podrá acceder posteriormente a estos datos históricos.



**Imagen 4-7: Diagrama de secuencias para las peticiones desde el gemelo digital al sistema de control del buque**

#### 4.4.4. Modelos de simulación

La capacidad de replicar el comportamiento de un sistema es una de las capacidades más importantes con las que debe contar un gemelo digital (C. Cabos & Rostock, 2018; Erikstad & Ove, 2017). Es la funcionalidad clave que permite pasar de la maqueta digital al gemelo digital. El uso de modelos de simulación permite pasar de lo que hasta ese momento puede ser considerado como un modelo virtual con capacidad de monitorización en tiempo real, a ser un gemelo, el cual permite comparar los datos reales con los datos esperados, o incluso generar señales no monitorizadas (lo que se conoce como sensores virtuales) a través, precisamente, de esos modelos de simulación.

Como se detalló en el capítulo 3, en general, a la hora de hablar de modelos de simulación en el entorno del gemelo digital, se puede hacer referencia a: modelos basados en física y modelos basados en datos. Debido a la inexistencia de datos experimentales en el momento en el que se comenzó a trabajar en los modelos de simulación del sistema HVAC, se consideró la aplicación de modelos de simulación basados en física. Una de las principales ventajas que presenta el uso de una herramienta de simulación de sistemas multifísicos es que permite modelizar de forma independiente cada uno de los componentes que forman el sistema, mediante la aplicación de las ecuaciones físicas que determinan su comportamiento. Después estos componentes pueden ser conectados entre sí mediante los enlaces correspondientes para que el sistema, en su conjunto, pueda ser simulado.

Este tipo de herramientas permiten desarrollar modelos 1D multidominio, que buscan simular el comportamiento de un sistema, y que son capaces de cubrir diferentes disciplinas y su integración entre ellas. Pueden incorporar, además, la simulación del control del sistema. Dentro de estos tipos de herramientas se encuentran: MATLAB SIMULINK, AMESIM, o por ejemplo aquellas herramientas que se basan en el lenguaje Modelica (OpenModelica, Dymola, Maplesim). Estas herramientas permitirán, en el entorno del desarrollo de un gemelo digital, simular sistemas de forma más o menos simplificados a partir de esquemas funcionales y/o especificaciones de los equipos que componen el sistema real. Además, todas estas herramientas permitirán exportar los modelos de simulación a un formato estándar (FMI) que permita ejecutar modelos sin necesidad de disponer del software con el que se crearon los modelos.

Aunque existen desarrollos relacionados con todas las herramientas citadas, analizada la bibliografía disponible se observó que muchos de los avances relacionados con modelos de simulación para gemelo digital consideran software basado en el lenguaje Modelica. Vöth, Magargle y Madni, presentan diferentes ejemplos de desarrollos para gemelos digitales que incorporan modelos de simulación basados en el lenguaje Modelica (Madni et al., 2019; Magargle et al., 2017; Vöth & Vasilyeva, 2020). Además de ello, se valoró la condición opensource de la herramienta OpenModelica y la facilidad con la que con ella se pueden exportar modelos a formato FMU. Por lo que en esta etapa inicial de desarrollo se optó por seleccionar esta herramienta. Modelica

es un lenguaje orientado a objetos y basado en ecuaciones, desarrollado para describir de forma sencilla modelos de sistemas físicos eventualmente complejos. Se trata de uno de los lenguajes más ampliamente empleados para la simulación de sistemas multi-físicos. Existen numerosas bibliotecas desarrolladas para representar una gran variedad de sistemas físicos diversos. Se resumen a continuación los modelos que se desarrollaron en OpenModelica.

#### 4.4.4.1. Modelos iniciales para simulación en tiempo real.

La primera iteración a la hora de desarrollar los modelos de simulación a implementar en el gemelo digital, y dada la fase de desarrollo del piloto (información del sistema físico muy escasa), se centró en la elaboración de unos modelos preliminares de caracterización de comportamiento, basados en ecuaciones sencillas que trataban de describir el comportamiento de los componentes del sistema. Las variables y las ecuaciones empleadas para estos modelos iniciales se incluyen en la tabla correspondiente al modelo de datos del apéndice A.

Estas ecuaciones se programaron en OpenModelica para su posterior exportación a formato FMU. Se ha de destacar que se han podido desarrollar modelos de simulación sencillos empleando únicamente código (sin necesidad de definir los característicos bloques en OpenModelica). Solo ha sido necesario definir en el código los parámetros que cada FMU debe leer del sistema real, las variables de salida que se van a calcular y devolver a la plataforma del gemelo y las ecuaciones que servirán para simular los componentes. Aunque una de las principales ventajas del software de este tipo es, precisamente, que permite caracterizar componentes físicos de aquellos que están en sus librerías, de forma que no es necesario escribir código, OpenModelica sí lo acepta, y a la hora de establecer funciones matemáticas, de hecho, se recomienda desde la propia plataforma.

#### 4.4.4.2. Modelos de predicción de comportamiento

Posteriormente se realizaron 2 modelos de simulación completos del sistema: uno para el modo verano y otro para el modo invierno, con el objetivo de poder predecir el comportamiento del sistema al completo.

**Sistema en modo invierno (calefacción):** Cuando el sistema trabaja en este modo, no hay circulación de agua en la parte hidráulica del sistema.

Simplemente se enciende una resistencia eléctrica a través de la cual pasa el aire (de forma que se calienta) que es impulsado por un ventilador. De modo que el funcionamiento del sistema se puede resumir en lo siguiente en un modelo 1D.

- RoomVol es el volumen de la habitación que se quiere calentar.
- Resistance: resistencia eléctrica que se encuentra en el interior del unitcooler y que sirve para calentar el aire.
- Fan: ventilador.

Para modelar el sistema se emplearon componentes de la librería de OpenModelica, Buildings (Perera et al., 2016; Wetter, 2018; Wetter et al., 2011). Se describen a continuación los principales componentes empleados en el modelo:

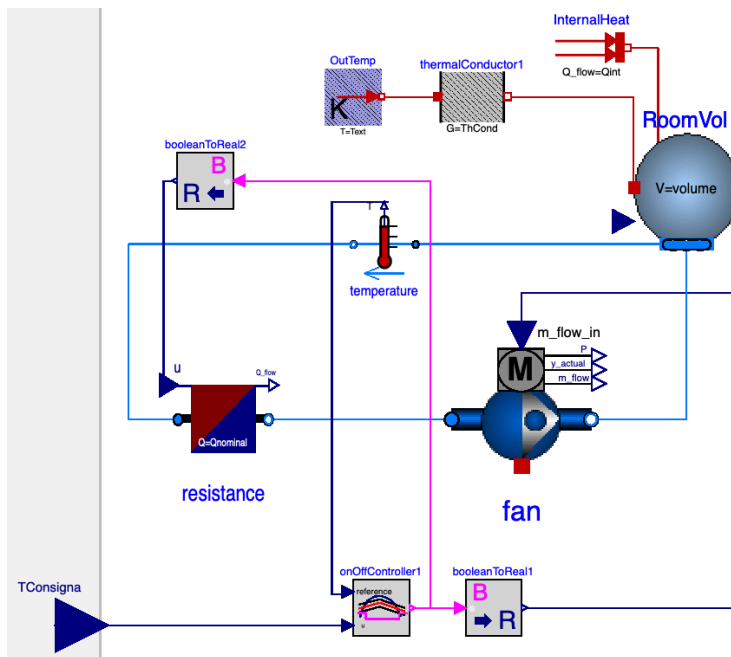


Imagen 4-8: Esquema del modelo desarrollado para modo invierno

**RESISTANCE:** Modelo para un calentador o enfriador ideal con flujo de calor prescrito al medio. Este modelo agrega calor en la cantidad:

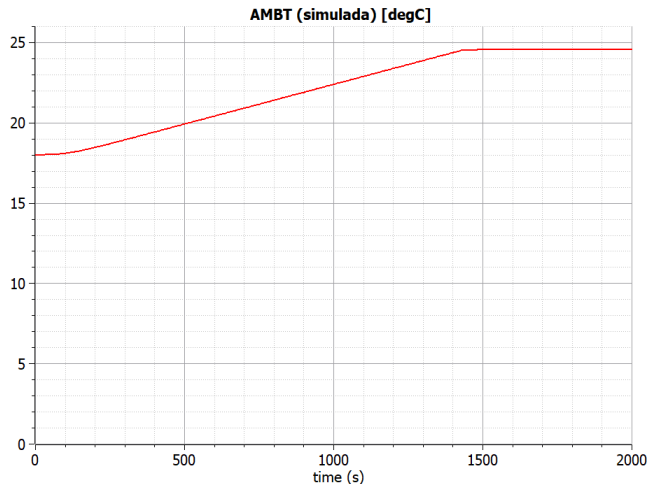
$$Q_{flow} = u Q_{flow\ nominal} \quad (4-1)$$

al medio. Este modelo no afecta la humedad del aire.

**FAN:** Este modelo describe un ventilador con caudal másico prescrito. La eficiencia del ventilador se calcula en base a las curvas de eficiencia y presión (si no se dispone de ellas, se puede seguir empleando el componente, referido únicamente un valor de caudal nominal).

**ROOMVOL:** Este modelo representa un volumen de control en el que el medio se mezcla de forma instantánea. Se desprecian la energía potencial y cinética en el puerto y no hay caída de presión en los puertos. El volumen puede intercambiar calor a través de su heatPort. En este caso se planteó que en ese puerto térmico se incorporara el calor generado por el equipamiento que se encontraba dentro de la habitación, así como la transferencia de calor con el espacio exterior a la habitación.

Este modelo inicial simula también el control del sistema: la resistencia y el ventilador funcionarán hasta que la temperatura de la cabina sea superior o igual a la temperatura de consigna + 2°C. La simulación de los modelos parte de un estado inicial en el que el sistema está apagado, y únicamente tiene como entradas la temperatura inicial y la temperatura objetivo. En la imagen 4-9 se muestran los resultados obtenidos para una simulación en la que se parte de una temperatura de la habitación de 18°C y se marca una temperatura objetivo de 22°C:



**Imagen 4-9: Evolución de la temperatura de la habitación (AMBT) de acuerdo con el modelo para calefacción desarrollado en OpenModelica**



**Sistema en modo verano:** en este modo, el modelo se complica, ya que debe incorporar toda la parte hidráulica, por lo tanto, estará formado por lo siguiente:

- RoomVol es, al igual que en el modo invierno, el volumen de la habitación que se quiere calentar.
- Fan: ventilador definido por sus valores nominales.
- Tank: Tanque de almacenamiento de agua.
- CentrifugalPump: Bomba centrífuga.
- Val: Válvula reguladora de caudal.
- Cooler: Intercambiador de calor agua/aire.

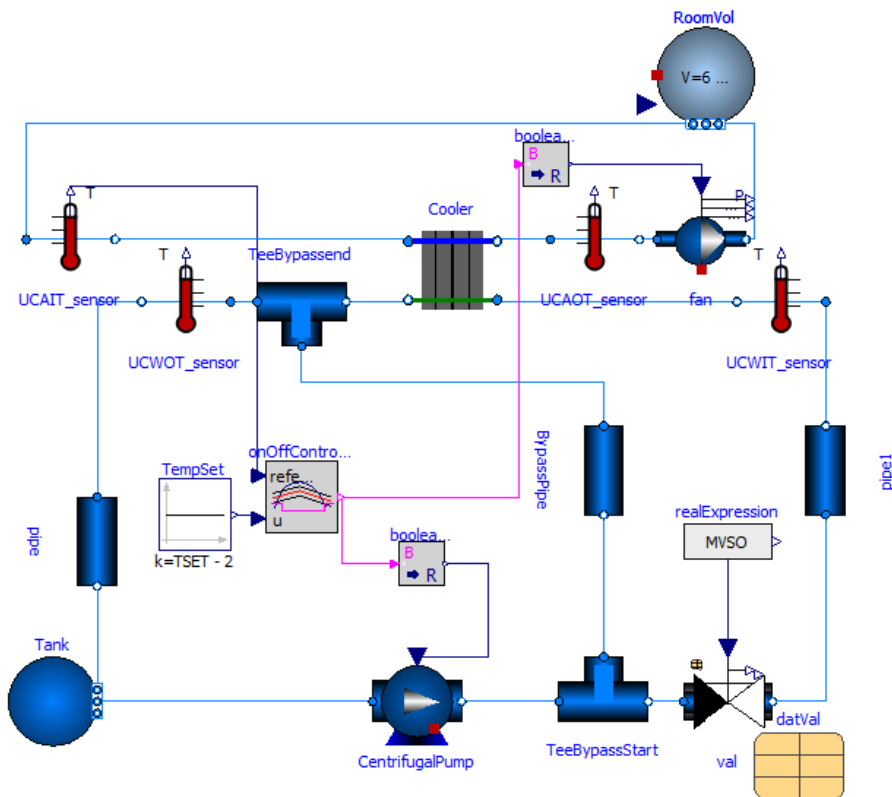


Imagen 4-10: Esquema del modelo desarrollado para modo verano

En este caso el modelo incorpora el sistema hidráulico (bomba, válvula, intercambiador de calor agua/aire), además del ventilador y el volumen de control ("RoomVol"). En este caso, la bomba y el ventilador funcionarán siempre que la Temperatura de consigna sea mayor que  $AMBT + 2\text{ }^{\circ}\text{C}$ . Al igual que

en el modo anterior, la simulación de los modelos parte de un estado inicial en el que el sistema está apagado, de modo que las entradas, de nuevo, son la temperatura inicial y la temperatura objetivo. En realidad, y como se verá más adelante (cuando se defina el modelo desarrollado en AMESIM) el control del sistema en modo verano es más complejo de lo que aquí se propone. En el momento en el que se realizó este modelo, aún se estaba trabajando en el diseño del control del sistema real, por lo que se optó por partir del caso más sencillo, en el que el ventilador trabajaba a revoluciones constantes.

Para este modelo se han combinado componentes pertenecientes a la librería *buildings*, con componentes de las librerías estándar de *OpenModelica* como por ejemplo *Modelica.fluids* (Kofman, n.d.):

**CENTRIFUGAL PUMP:** Este modelo describe una bomba con velocidad prescrita, ya sea fija o proporcionada por una señal externa (en este caso las revoluciones de la bomba son constantes: 2900 rpm). En los parámetros de la bomba es posible asignarle la presión en la descarga, que en este caso se fijó en 2 bar.

**VAL:** Válvula reguladora de caudal de 2 vías. El caudal de la válvula en función del porcentaje de apertura de la válvula se establece a partir de los datos suministrados en una tabla. Además, es necesario definir el coeficiente  $K_v$  de la válvula.

**COOLER:** Modelo de intercambiador de calor de eficacia constante. Este modelo transfiere la siguiente cantidad de calor

$$Q = \varepsilon Q_{m\acute{a}x} \quad (4-2)$$

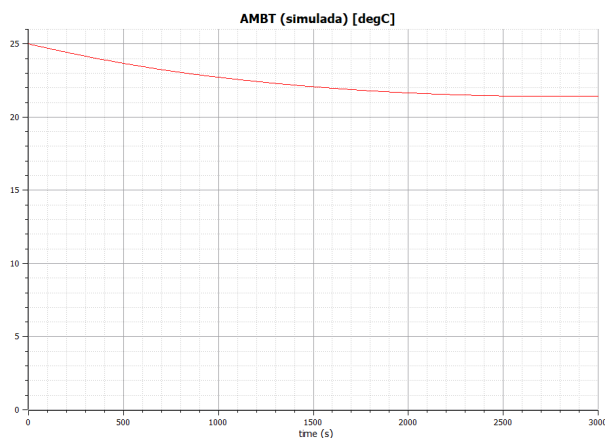
Donde  $\varepsilon$  es la eficiencia (constante) y  $Q_{max}$  es el calor máximo que podría ser transferido. En este componente simplemente se requiere definir esa eficiencia, así como los caudales y las caídas de presión (nominales) en el lado agua y en el lado aire.

**FAN:** Este modelo describe un ventilador con caudal másico prescrito. La eficiencia del ventilador se calcula en base a las curvas de eficiencia y presión (si se dispone de ellas).

**ROOMVOL:** Este modelo representa un volumen de control en el que el medio se mezcla de forma instantánea.

**TANK:** Este componente representa un volumen de agua, con una temperatura y presión fijadas. La temperatura se considera constante, a 11°C, tratando de representar el enfriamiento del agua en la unidad de refrigeración de agua del sistema.

Se llevaron a cabo simulaciones empleando este modelo, obteniendo los siguientes resultados (imagen 4-11):



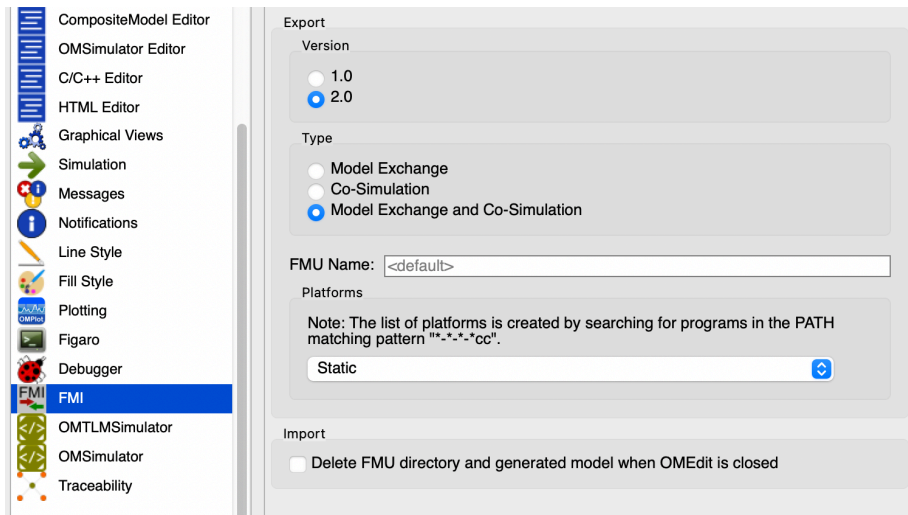
**Imagen 4-11: Evolución de la temperatura de la habitación (AMBT) de acuerdo con el modelo para modo verano desarrollado en OpenModelica**

#### 4.4.5. Modelos ejecutándose en la capa de gemelo digital

La ejecución de los modelos de simulación en el entorno del gemelo digital se llevó a cabo exportando los modelos desarrollados en Openmodelica a formato FMU para poder ser ejecutados desde el gemelo digital.

La exportación de modelos de simulación a formato FMU desde el software OpenModelica es muy sencilla, se realiza a través de una ventana gráfica que no s permite seleccionar la versión del estándar que queremos: 1.0 o 2.0 y si queremos un modelo model exchange, co-simulation, o ambos.

La diferencia entre estos dos tipos de exportación es la siguiente: cuando se exporta un modelo como Co-simulation, el FMU incorporará el solver necesario para que ese FMU pueda ser ejecutado en la herramienta en la que se importe, mientras que, si el FMU se exporta como Model Exchange, no incorporará el solver, de modo que la herramienta en la que se importe tendrá que usar su propio solver para ejecutar ese modelo.



**Imagen 4-12: Interfaz para configurar la exportación a FMU desde OpenModelica**

Una vez seleccionadas esas características, se puede exportar el modelo y se obtendrá un archivo con extensión “.fmu”. Este archivo en realidad se trata, como se ha comentado, de un fichero comprimido, que puede ser procesado como un solo archivo a la hora de ser importado en otra plataforma, o a la hora de ser ejecutado desde el entorno seleccionado para la ejecución de los modelos dentro del gemelo digital, pero en realidad, como ya se ha descrito en detalle en el capítulo 3 se compone de lo siguiente (Fritzson, 2011):

- Una carpeta denominada binaries que incluye las ecuaciones del modelo en código fuente (en C) o binarios (dll) para una o más plataformas (Windows, Linux...).
- Un fichero denominado modelDescription.xml. Se trata de un fichero xml que contiene información del modelo: variables, tipos de datos, parámetros por defecto...
- Una carpeta denominada resources, con otro tipo de información que pueda ser necesaria para la ejecución del modelo: documentación en formato html, icono para el modelo, mapas, tablas...



Imagen 4-13: Componentes internos de un paquete FMU

Para la ejecución de los modelos de simulación, en formato FMU desde el gemelo digital, se optó por emplear la librería de Python PyFMI, de JModelica (Andersson et al., 2016; Modelon AB, 2018; Workshop & Angeles, 2015). En primer lugar, fue necesario llevar a cabo la instalación de lo siguiente: Distribución de Python Anaconda: se instaló la versión de Python 2.7 en un paquete completo. Como paquetes Python (con la herramienta pip) se instalaron las librerías que son dependencias para la ejecución de modelos a través de una API REST: librería PyFMI (para la ejecución de FMU), librería Falcon (para poder emplear API REST) y la librería Waitress (servidor de aplicaciones). Se puede denominar a este conjunto como servidor de simulaciones. El servidor de simulaciones contiene los FMU y se encarga de ejecutar las simulaciones en base a las peticiones que realiza el gemelo digital. Un FMU es una caja negra (a no ser que se proporcione información adicional) que contiene el código compilado que permite simular el sistema físico.

En las peticiones de simulación se contienen los parámetros necesarios para la simulación, como, por ejemplo: tiempo final de la simulación, número de puntos calculados, parámetros de entrada. El servidor de simulaciones se puede ejecutar en el mismo servidor físico en el que se encuentra Thingworx o en otro servidor diferente. Los requisitos para ejecutar las simulaciones son Python, la librería PyFMI y la librería de servicios web Falcon: la librería PyFMI es necesaria para cargar y ejecutar los FMU y la librería Falcon es

necesaria para proporcionar la API REST de forma sencilla. El servicio de simulaciones se puede ejecutar en un puerto configurable y escucha constantemente a peticiones HTTP por parte del gemelo digital. En caso de ejecutarse el servidor de simulaciones en otro servidor físico, debe existir una comunicación a nivel de red, por ejemplo, Ethernet, entre gemelo digital y servidor de simulaciones.

Una vez que se solicita al servidor de simulaciones la ejecución de una simulación, la simulación finaliza, se devuelven los datos calculados y el servidor de simulaciones se queda esperando por una nueva petición de simulación. De esta forma, es responsabilidad del gemelo solicitar la ejecución de modelos al servidor de simulaciones. En Thingworx, existen temporizadores de eventos para ejecutar determinadas piezas de código de forma periódica. En este caso, la ejecución de las simulaciones y la petición de señales al sistema de control se haría de esta forma. Se muestran a continuación los diagramas de secuencias donde se define la llamada desde el gemelo digital al servidor de simulaciones, y el diagrama de actividades para ejecución de los FMU.

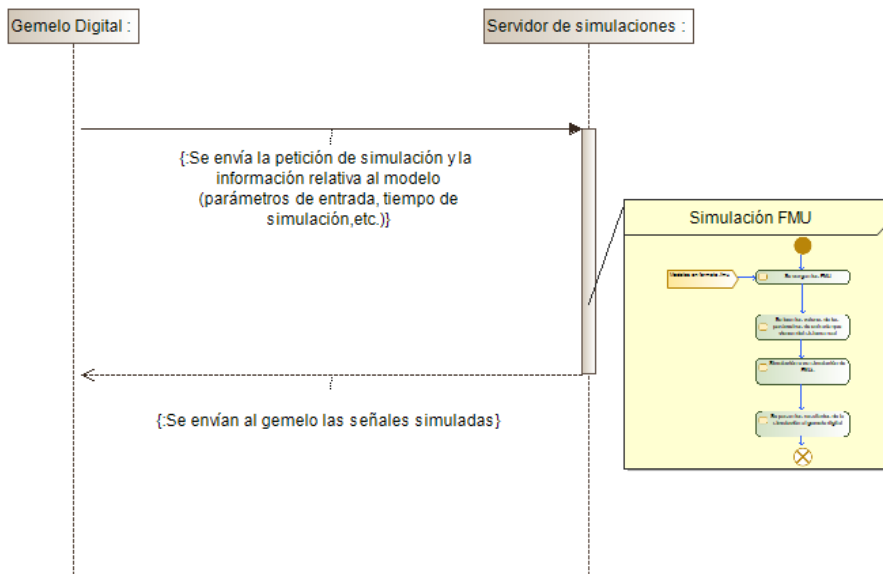
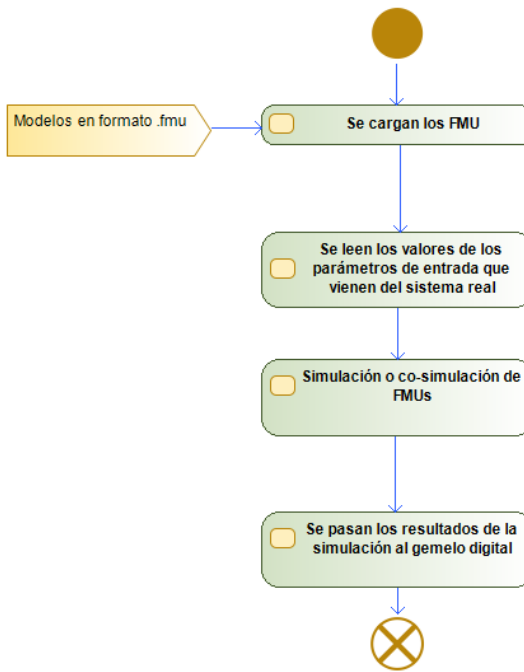


Imagen 4-14: Diagrama de secuencia para ejecutar simulaciones desde el gemelo digital



**Imagen 4-15: Diagrama de actividades: Ejecución de FMU en servidor de simulaciones, librería PyFMI**

Según el componente y el FMU que se simula, las variables generadas en la simulación son diferentes, pero existe una estructura de datos con todas las variables para devolver siempre esta estructura al ejecutar una simulación (de acuerdo con el modelo de datos establecido). Según el componente que se simule, unas variables se devolverán mientras que las otras irán vacías. En el caso de los modelos de simulación en tiempo real el resultado será un número “n” de variables en un determinado tiempo, mientras que, en el caso de los modelos para predicción de comportamiento, el resultado será una matriz con la evolución de las n variables a lo largo del tiempo de simulación

Los resultados de las simulaciones se almacenan en el gemelo digital, para poder ser visualizados, ya sea el caso de las simulaciones en tiempo real, como en el caso de las simulaciones para predicción de comportamiento (que como se detallará a continuación, no se procesan de la misma forma). En el siguiente apartado, correspondiente al almacenamiento de datos, se explica cómo se almacenan en Thingworx los datos de las simulaciones (además de los datos recibidos del sistema físico a través del sistema de control).

#### 4.4.6. Almacenamiento de datos

Los valores de las señales del sistema real que se reciben por API-Rest desde el sistema de control se almacenan en Thingworx como un elemento de tipo “stream”(PTC, 2019b). Las entidades de tipo stream sirven para representar series de datos. En este caso, cuando Thingworx recibe el fichero JSON con los datos, recorre este fichero y almacena cada variable en su correspondiente stream, con una etiqueta de tiempo (que es fijada por Thingworx). Este retraso en el etiquetado de tiempo no reviste importancia ninguna en el caso de uso que se plantea (ya que el gemelo digital solicita los datos al sistema de control cada 2 segundos, de modo que el retraso que pueda existir desde que se mandan los datos solicitados hasta que Thingworx los almacena, será de fracción de segundo). Aunque es importante señalar que esto sucede así, ya que los datos llegan al gemelo digital sin etiqueta de tiempo desde el sistema de control.

Algo similar se planteó en la ejecución de los modelos de simulación en tiempo real, en este caso, de nuevo se lanza una petición desde el gemelo digital al servidor de simulaciones, el servidor de simulaciones devuelve un valor de las variables simuladas, que Thingworx recibe y guarda en el stream correspondiente (fijando en este momento su etiqueta de tiempo). Es evidente que también habrá un retraso (ínfimo) en el etiquetado de la señal.

En el caso de que la simulación que se ejecute sea de predicción de comportamiento, los datos se almacenan de forma diferente. No es posible almacenarlos como stream. La estructura para almacenar esto en Thingworx es a través de tablas, lo que en Thingworx se denomina DataTable (PTC, 2019c), con lo cual previamente fue necesario crear la estructura de esta tabla de modo que se rellena cada vez que se ejecuta la simulación.

Las instancias de Thingworx se almacenan internamente en bases de datos relacionales. En el caso de uso planteado, se usó una base de datos H2, al tratarse de una prueba de concepto. H2 es una base de datos relacional de código abierto con una huella de disco reducida y escrita en Java (PTC, 2019a). Este es el modo de conexión más rápido y fácil. Se trata de una base de datos persistente, por lo que los datos se escriben en el disco local (en la carpeta de la base de datos en ThingworxStorage, el directorio que establece Thingworx para almacenamiento). Se trata de una base de datos pensada para desarrollar



pruebas de conceptos, por lo que presenta limitaciones importantes si se quiere emplear en un sistema real plenamente funcional, debidos precisamente a la característica principal de esta base de datos, se encuentra integrada dentro de la aplicación, por lo que: tiene una escalabilidad limitada ya que comparte recursos con la aplicación, puede comprometer la estabilidad de la aplicación en su conjunto.

En aplicaciones reales, Thingworx propone emplear como bases de datos internas a la aplicación bases de datos PostgreSQL y DataStax Enterprise Cassandra (y más recientemente InfluxDB). De forma externa a Thingworx, se puede usar prácticamente cualquier tecnología de almacenamiento de datos que proporcione un conector JDBC o incluso uno que tenga un controlador que se pueda usar para crear una extensión Thingworx a través de un servicio SDK, de forma que se podrían emplear bases de datos externas como Oracle o SQL Server. Esta parte del desarrollo no se pudo probar en las pruebas que se realizaron con la plataforma Thingworx, pero en general, lo que se recomienda es emplear las bases de datos internas de Thingworx.

El almacenamiento de la plataforma Thingworx está optimizado para almacenar y recuperar datos de series de tiempo, utilizando los tipos ValueStream y Stream integrados en la plataforma. Esta es la estructura de datos de IoT más común consiguiendo además que estos datos sean muy fácilmente accesibles posteriormente, aún cuando se quiere consultar un volumen de datos importante (entre 100 000 y 200 000 señales). El uso de almacenamiento en entidades de tipo datatable (como las empleadas para guardar los resultados de los modelos de predicción de comportamiento) solo se recomienda cuando el tamaño no vaya a ser muy grande (alrededor de 100 000 registros o menos),

La consulta y recuperación de datos es comúnmente donde se ven los mayores problemas de rendimiento.

En el caso de las bases de datos externas, tampoco se recomienda emplearlas para almacenamiento de históricos de los que se quieran hacer frecuentemente consultas grandes, se recomienda emplear estas bases de datos para otro tipo de información que deba contener el gemelo.

#### **4.4.7. Conexión con sistemas de información**

La decisión tomada por Navantia de cambio de plataforma, hizo que parte de los desarrollos programados para ser llevados a cabo en el entorno

Thingworx, no pudieran realizarse. Entre ellos, la conexión con el PLM, Windchill, mediante el cual se esperaba realizar la conexión del gemelo digital con el gestor de ciclo de vida del buque, aspecto imprescindible en el desarrollo de gemelos digitales en buques.

Si bien, la conexión de Thingworx con los Sistemas de Información del buque, y en concreto, con el PLM Windchill y el Gestor de recursos empresariales, SAP, se validó en los desarrollos realizados dentro de la línea de IoT de la UMI (por el mismo equipo de trabajo que la línea de gemelo digital), de modo que se incluye un pequeño resumen de realizado en esta línea referentes a la conexión con el PLM. Para llevar a cabo la comunicación entre ThingWorx y Windchill fue necesaria la instalación del paquete ThingWorx Navigate. Éste es un conjunto de aplicaciones que permiten acceder fácilmente a los datos actuales del producto desde múltiples fuentes. El acceso está basado en roles para que cada grupo pueda obtener la información que sea significativa desde su punto de vista. ThingWorx Navigate puede extraer datos del PLM, pero también de otros sistemas instalados en la empresa. Las funcionalidades que se pueden usar son:

- Ver requisitos.
- Ver archivos de diseño.
- Ver dibujo.
- Ver y medir en 3D.
- Ver propiedades de la pieza.
- Ver lista de piezas.
- Ver estructura de la pieza.

Se crearon una combinación de servicios que permite mostrar el árbol de archivos del PLM (a partir del identificador denominado "Part Number" se obtiene la configuración de padre-hijos) y seleccionar un archivo para ver el modelo 3D correspondiente. En la imagen 4-16 se muestra cómo se realiza el envío de datos entre los servicios y los "widgets" (elementos para mostrar información en las interfaces). Cada servicio devuelve habitualmente un "Info-table" (tabla de almacenamiento de datos), que se asocia al widget correspondiente, donde se muestran total o parcialmente los datos almacenados en él.

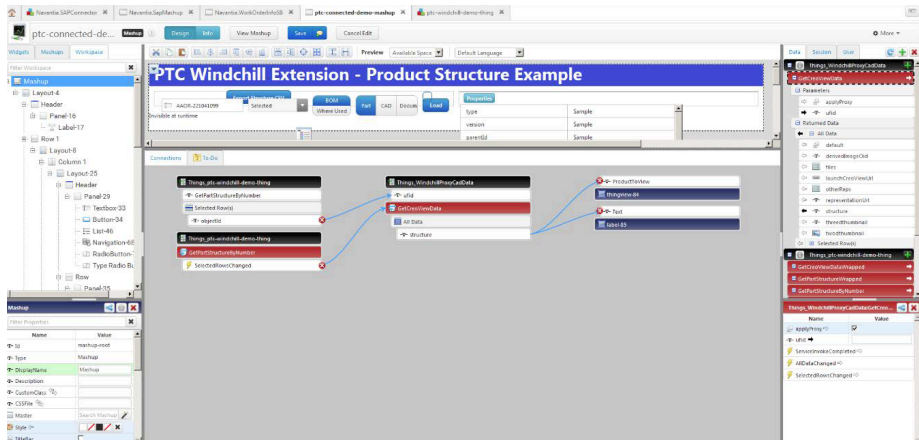


Imagen 4-16: Conexiones entre los widgets de las interfaces y los servicios

Respecto a la conexión de Thingworx con SAP, esta se realizó a través de servicios ODATA, un protocolo que permite hacer APIs RESTful, pero fue más compleja de lo esperado inicialmente. La especificación ODATA es un estándar abierto que puede ser implementado por diferentes programas y, en este caso, debían implementados por el SAP Netweaver Gateway y por el conector de SAP integrado en ThingWorx (Integration Runtime). La codificación de los datos para estos servicios ODATA se realiza en XML o en JSON. A través de ODATA se suelen proveer cuatro funcionalidades para cada tipo de entidad:

- Crear entidad (http post).
- Leer entidad o conjunto de entidades (http get).
- Actualizar entidad (http put).
- Borrar entidad (http delete).

Se hicieron pruebas para leer cabeceras y detalles de órdenes de trabajo de fabricación del buque:

- BAPI\_GET\_PRODORD\_LIST para obtener una lista de cabeceras de ordenes.
- BAPI\_GET\_PRODORD\_GET\_DETAIL para obtener los detalles de una orden de fabricación.

Se comprobó que, al realizar el servicio en base a estas funciones, había parámetros de entrada obligatorios a la llamada a la BAPI que no se podían configurar a través de ThingWorx. Además, los tipos de salida, debido a ser tipos complejos, tampoco se podían obtener desde ThingWorx. La solución

implementada en código propio de SAP, ABAP, fue la llamada interna a las BAPIs con los parámetros de entrada configurados en el código.

#### 4.4.8. Conclusiones

En la implementación del gemelo digital de un sistema HVAC basado en una plataforma comercial de IoT se pudo desarrollar y validar lo siguiente:

- Se implementó una alternativa para adquisición de datos desde el sistema real hacia el gemelo digital, a través del sistema de control del buque, basado en protocolos API-Rest.
- Se desarrollaron modelos de simulación en una herramienta de simulación de sistemas multi-físicos. Se llevó a cabo la exportación de los modelos de simulación al estándar FMI, y se validó la ejecución de estos modelos en el entorno del gemelo digital.
- Se analizaron diferentes alternativas para el almacenamiento de datos históricos en la plataforma de IoT.
- Aunque no se realizó en este caso de uso, se validó la conexión de la plataforma Thingworx con los sistemas de información, por un lado, la conexión con el gestor de ciclo de vida del buque, en la que se integró la visualización de modelos 3D, y por otro lado, la conexión con el ERP, para visualización de estado de órdenes de fabricación, en este caso, SAP.

Si bien, quedó pendiente de desarrollar lo siguiente, debido a que finalizó el tiempo disponible para pruebas con la plataforma Thingworx:

- Validar la adquisición de datos con el paquete Edge de PTC, sino a través del sistema de control, lo que es un condicionante distinto, ya que ThingWorx está pensado para realizar la adquisición de datos y su posterior procesado.
- Analizar las posibilidades que ofrece el paquete de Analítica de Thingworx.

Como principales ventajas identificadas en la implementación de un gemelo digital basado en una plataforma IoT cabe destacar lo siguiente:

- La arquitectura centralizada en la plataforma ThingWorx permitía la ejecución de los FMU directamente en el entorno de la

plataforma, por lo que se simplificó considerablemente la implementación de los modelos de simulación.

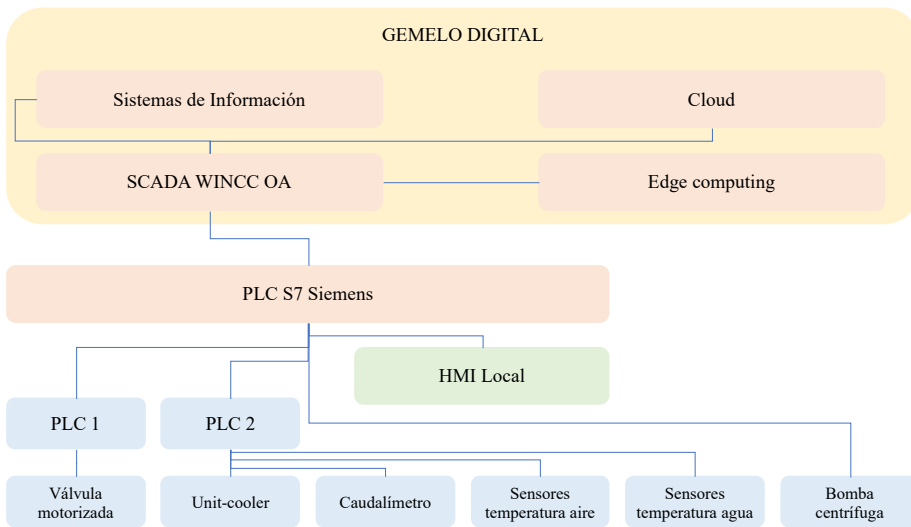
- La base de datos integrada en la misma plataforma tenía capacidad para almacenar los datos procedentes de los sensores y los datos obtenidos de las simulaciones, lo que permitía una consulta ágil y eficiente de los históricos para su despliegue en gráficas.
- La conexión de esta plataforma IoT con el PLM se realizó de forma sencilla, debido, principalmente, a tratarse de herramientas pertenecientes al mismo grupo.

En cuanto a los principales problemas o carencias identificadas en esta etapa:

- La conexión con SAP fue compleja y requirió de conocimientos específicos de programación de servicios ODATA y BAPIs.
- La licencia disponible para la plataforma IoT solo permitió usar bases de datos H2, no pudiendo analizar el uso de bases de datos PostgreSQL o InfluxDB.

#### **4.5. ARQUITECTURA DESCENTRALIZADA**

En la segunda arquitectura que se planteó en el desarrollo del gemelo digital del caso de uso HVAC, éste ya no se encuentra centralizado en la plataforma de IoT, sino que se encuentra en diferentes entornos interconectados. Se planteó un desarrollo centralizado en la capa local, en contraposición a lo planteado en la plataforma de IoT (que en general está orientado a la nube) y basado en el software de Siemens, como se detalla en la imagen 4-17: en este caso, sobre la capa física, se sitúa una capa de computación local, formada por lo que se denomina “edge computing” y un sistema SCADA. En este caso la parte de interfaz de usuario (visualización de señales de sensores, documentación, alarmas, etc. Se realiza a través del SCADA, mientras que en el edge se centraliza la ejecución de modelos numéricos, ya sean modelos de simulación, modelos de análisis de datos, machine learning, etc.).



**Imagen 4-17: Arquitectura descentralizada**

Sobre esa capa se pueden desplegar conexiones hacia la nube, en la que se podrían plantear diferentes desarrollos o la conexión con los sistemas de información, que en el caso anterior se realizaron, principalmente, con el PLM de PTC, Windchill y que en este caso se podrían desarrollar con el PLM de Siemens, Teamcenter, aunque estos dos últimos aspectos, no serán objeto de esta tesis doctoral. En este caso, el trabajo se ha centrado principalmente en el desarrollo de modelos de simulación y en su implementación en este nuevo entorno para el gemelo digital.

#### 4.5.1. Adquisición de datos hacia el gemelo digital

Como se ha comentado, la principal diferenciación de este segundo planteamiento se basa en la integración dentro de la arquitectura de una capa de edge computing. Esta integración implicó un cambio en el paradigma establecido de inicio, y en lo referente a la adquisición de datos afectará a que: en la parte superior de la capa de control tendremos un SCADA 4.0 y elementos para EDGE computing. Esto va a permitir mantener una latencia mínima en la adquisición de datos y en la respuesta después del procesado, ya sea a través de modelos de simulación del sistema, o de algoritmos de machine learning. Se puede decir que, en esta arquitectura, la capa de control pertenece al gemelo digital; en este caso no se centraliza el gemelo en torno a una plataforma de

IoT, sino que se dispone cada elemento en aquella ubicación óptima desde el punto de vista de la adquisición y procesado de señales (en la capa edge).

#### **4.5.2. Modelos de simulación**

Se desarrollaron modelos de simulación 1D, en AMESIM. En este caso, se plantearon 2 modelos: un modelo para ser ejecutado en tiempo real con el objetivo de poder comparar los resultados obtenidos con los del sistema real y otro modelo para predicción de comportamiento. Los componentes de ambos modelos son similares, las principales diferencias entre ambos radican en las entradas/salidas que estos pueden generar, y además de ello en que el modelo de predicción de comportamiento debe simular el sistema de control del sistema real para poder representar su comportamiento de forma fiable.

En este modelo se ha caracterizado la bomba centrífuga, la válvula motorizada, el intercambiador de tubos y aletas, la resistencia eléctrica y la unidad enfriadora. La caracterización de estos componentes se llevó a cabo combinando los datos aportados por los fabricantes, con pruebas realizadas en el sistema.

En la imagen 4-18 se presenta un esquema del modelo de simulación desarrollado. Para mejor visualización y comprensión del modelo, algunos componentes se han simplificado (solo de forma gráfica) como, por ejemplo, la habitación (que en realidad estará compuesta por las paredes que forman el espacio considerado y las fuentes de calor interno).

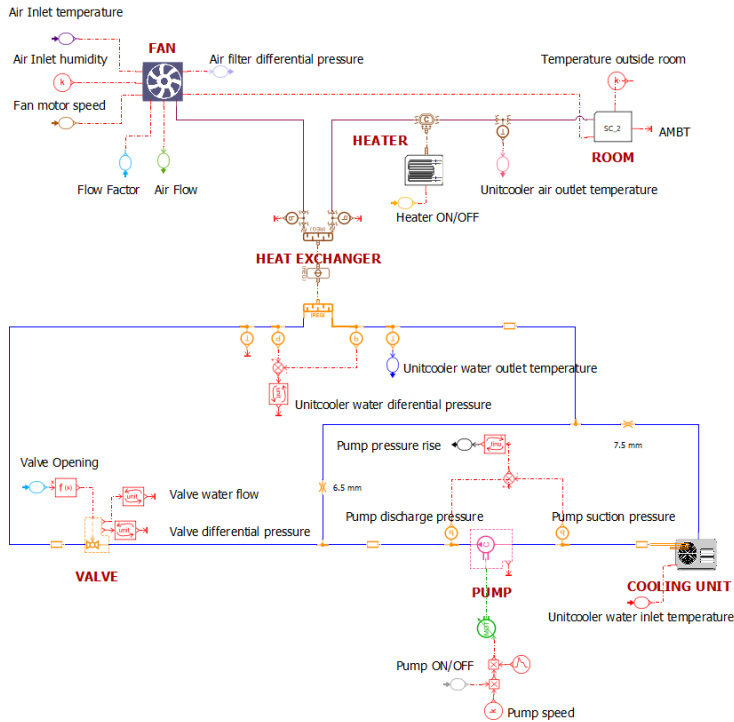


Imagen 4-18: Esquema del modelo de simulación multifísico

En el modelo se trató de ajustar el comportamiento hidráulico y térmico, modelando los siguientes componentes:

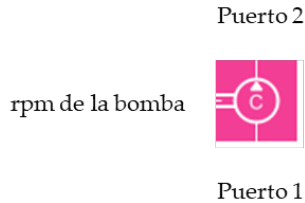
- Bomba centrífuga.
- Válvula.
- Intercambiador de calor.
- Calentador.
- Ventilador.
- Habitación.

A continuación, se describe brevemente en qué se basa el comportamiento de cada uno de los componentes de AMESIM. La información relativa a la base teórica se ha consultado en la documentación de la herramienta (Siemens-AMESIM, 2020c, 2020a, 2020b).

**Bomba centrífuga:** La bomba se ha modelado empleando un componente existente en AMESIM para bombas centrífugas termo-hidráulicas. El aumento



de presión del fluido que pasa a través de la bomba se calcula utilizando las leyes de afinidad. En este componente es necesario conocer el aumento de presión en los puertos de la bomba, en función del caudal volumétrico de salida de la bomba a una velocidad o más.



**Imagen 4-19: Componente para modelar una bomba centrífuga en AMESIM**

Se conocen las características de una bomba de referencia en condiciones de funcionamiento de referencia. De ellos se deduce el comportamiento del modelo de bomba. Las curvas características (datos en función del caudal volumétrico) se pueden proporcionar para una velocidad de referencia especificada (tablas 1D), para diferentes velocidades (tablas múltiples 1D) o adicionalmente para diferentes temperaturas (tablas múltiples 1D). En este caso, la bomba gira a rpm constantes, por lo que solo se definió una curva 1D para el incremento de presión/caudal.

El caudal másico en el puerto de entrada se calcula de modo que la presión de salida (alta presión) alcance el siguiente valor

$$p_{out} = p_{in} + \Delta p \quad (4-3)$$

Donde  $p_{in}$  es la presión de entrada (baja presión) y  $\Delta p$  es la diferencia de presión. La diferencia de presión en la bomba se interpola gracias a los datos de referencia. La temperatura y la presión a la salida se calculan realizando balances de masa y energía en este elemento, calculando las derivadas del tiempo de presión y temperatura como funciones de los flujos de masa entrantes y los flujos de calor transmitidos a través de los puertos del elemento (Siemens-AMESIM, 2020c). La potencia mecánica suministrada por la bomba al fluido se calcula del siguiente modo:

$$P_{mech} = \frac{Q \cdot \Delta p}{f_{eff}} \quad (4-4)$$

$$0 \leq f_{eff} \leq 1 \quad (4-5)$$

donde  $Q$  es el caudal volumétrico,  $\Delta p$  es la diferencia de presión y  $f_{\text{eff}}$  es la eficiencia global. El método de interpolación utilizado para obtener la diferencia de presión y la eficiencia (si es necesario) dependerá del modo de caracterización elegido, en este caso el método empleado para la caracterización se basa en las leyes de afinidad. Se asume que se conoce el comportamiento de una bomba específica en condiciones específicas y a partir de ahí, se puede deducir el comportamiento de una bomba similar en condiciones similares. Los valores de caudal volumétrico, presión, y potencia o velocidad se multiplican respectivamente por los siguientes coeficientes:

$$C_q = \frac{1}{N \cdot D^3} \quad (4-6)$$

$$C_p = \frac{1}{\rho \cdot (N \cdot D)^2} \quad (4-7)$$

$$C_w = \frac{1}{\rho \cdot N^3 \cdot D^5} \quad (4-8)$$

donde  $N$  es la velocidad de rotación de referencia,  $D$  el diámetro de la bomba de referencia, y  $\rho$  la densidad del fluido de referencia. Como resultado, se obtienen splines re-escalados. Durante la simulación, se realiza una interpolación en estas splines. Por ejemplo, el caudal volumétrico actual se multiplica por  $C_q$  (siendo  $N$  la velocidad actual y  $D$  el diámetro de la bomba simulada) y se obtiene un aumento de presión a partir de la interpolación.

**Válvula:** El componente empleado para modelar la válvula, es un componente de válvula de bola termo-hidráulica. En ella, se calculan en cada puerto la presión de entrada y los caudales de masa y entalpía. La señal de entrada en el puerto 1 se interpreta como grados y corresponde al ángulo de apertura de la válvula,  $\varphi$ . Para definir la característica de la válvula de bola, hay varios modos de configuración, el que se ha empleado en el modelo se basa en la tabla de coeficientes  $K_v$  de la válvula, en la tabla se define para cada grado de apertura de la válvula, el coeficiente  $K_v$  correspondiente.



Imagen 4-20: Componente para modelar una válvula en AMESIM

El coeficiente de capacidad de flujo máximo,  $Kv(\varphi)$ , relaciona el área de la sección transversal de la válvula y el coeficiente de flujo máximo,  $Cq(\varphi)$ :

$$Kv(\varphi) = 0.865 \cdot Cv(\varphi) \quad (4-9)$$

$$\text{Área}(\varphi) \cdot Cq(\varphi) = 1.6982 \cdot 10^{-5} \cdot Cv(\varphi) \quad (4-10)$$

$$Cq(\varphi) = \frac{1}{\sqrt{\xi(\varphi)}} \quad (4-11)$$

Donde  $\xi(\varphi)$  es el coeficiente de pérdida de carga singular de la válvula. La evaluación de caudal se determina a partir de:

$$dm = \rho \cdot Cq \cdot \text{área}(\varphi) \cdot \sqrt{\frac{2\Delta P}{\rho}} \quad (4-12)$$

**Intercambiador de calor:** El intercambiador de calor agua aire de aletas y tubos, se modeló empleando los componentes de AMESIM denominados Half-Exchanger y el componente Heater Flux.



**Imagen 4-21: Componentes para modelar un intercambiador de calor en AMESIM**

En realidad, lo que se hace en este caso es simular de forma separada los componentes del intercambiador (por un lado, el componente agua y por otro lado el componente aire) de forma que se definen parámetros como el volumen de fluido, caracterización de las pérdidas de carga, etc. para cada uno de los 2 “half-exchangers”. En un componente independiente (conectado entre ambos) se simula la transferencia de calor entre los 2 medios. Se calcula empleando el método NTU, y la forma de caracterizar el comportamiento térmico del intercambiador fue (dado que se disponía de datos) hacerlo a partir de datos reales, empleando una herramienta disponible en AMESIM, que permite hacer regresiones para determinar el comportamiento del intercambiador (HHX Regression Tool App).

La NTU Regression Tool es una herramienta que calcula la conductancia total UA utilizando el método de eficiencia-NTU.

$$\frac{1}{UA} = \frac{1}{UA_1} + \frac{1}{G_{pared}} + \frac{1}{UA_2} \quad (4-13)$$

donde los subíndices 1 y 2 representan el lado del intercambiador de calor. Se utilizan correlaciones térmicas simplificadas sin geometría para calcular las conductancias generales en ambos lados de los intercambiadores de calor dados.

$$UA_1 = a_1 \cdot dm_1^{b_1} \quad (4-14)$$

$$UA_2 = a_2 \cdot dm_2^{b_2} \quad (4-15)$$

$a_1$ ,  $b_1$ ,  $a_2$  y  $b_2$  son coeficientes que se determinan usando una regresión clásica de mínimos cuadrados en datos experimentales dados.

En la imagen 4-22 se muestra una visualización de la herramienta. Para poder emplear esta herramienta es necesario conocer el tipo de intercambiador (el medio que hay en cada uno de los lados, y cómo es el flujo: en este caso, transversal). Y posteriormente, se deben seleccionar los parámetros que se desean ajustar: la conductancia térmica de la pared y las cuatro correlaciones térmicas simplificadas.

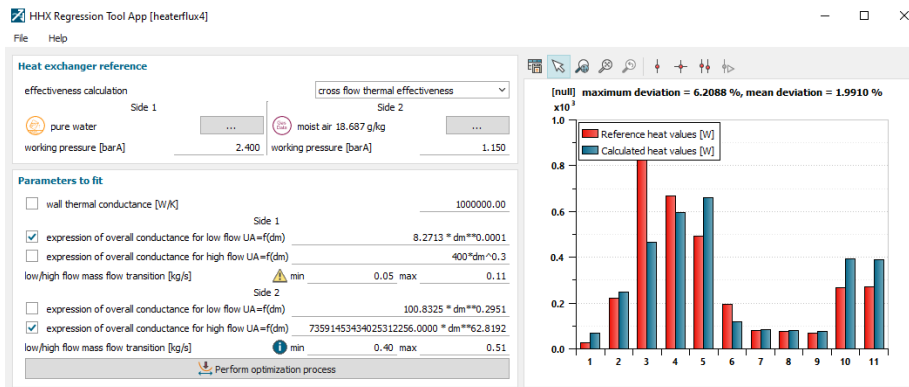


Imagen 4-22: Interfaz de la herramienta de regresión

La caracterización del intercambiador empleando este componente se completa con datos experimentales. En la herramienta es necesario introducir los datos de funcionamiento, tal y como se muestra como ejemplo en la tabla 4-1:

**Tabla 4-1: Datos experimentales necesarios como entradas para la herramienta HHX Regression Tool**

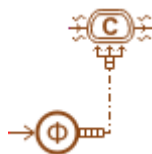
Experimental data					
	q1	tempIn1	q2	tempIn2	tempOut1
	L/h	degC	m <sup>3</sup> /h	degC	degC
1	421	7.97	1143	22.07	13.69
2	254	6.21	882	19.8	12.08
3	167	10.81	882	19.3	15.01
4	104	11.1	879	19.03	15.54
Fila n	-	-	-	-	-

Donde:

- q1 y q2 son los caudales (en l/h) de los 2 medios (agua/aire).
- tempIn1 y tempIn2 son las temperaturas de entrada de ambos medios.
- TempOut1 es la temperatura a la salida del agua.

Con todo lo anterior, se puede ajustar el intercambiador de calor, con lo que el componente quedaría caracterizado para los datos de referencia.

**Calentador.** El calentador eléctrico del unit cooler se simuló empleando una fuente de calor y una cámara de aire en la que se aplica ese calor.



**Imagen 4-23: Componentes para simular un flujo de calor y un volumen de control (cámara de aire)**

Esta combinación permite simular de forma aproximada el aporte de calor que se produce desde la resistencia eléctrica al volumen de aire en el interior del unit cooler, considerando el caudal de aire que circula por el mismo. Es importante tener en cuenta, que en el momento en el que se desarrolló el modelo, no se tenía demasiada información respecto al unit cooler, más allá de los resultados que se obtenían de las señales de los sensores. Esto limitó el desarrollo de esta parte del modelo.

**Ventilador.** Debido al desconocimiento de las características en detalle del motor (solo se contaba con los valores nominales), el ventilador también se simuló de forma simplificada. El fabricante aportó una relación directa entre las revoluciones del ventilador y el caudal de aire suministrado, por lo cual se decidió simular el ventilador como una fuente de aire a la temperatura y humedad de entrada en el unit cooler.



Imagen 4-24: Componente para simular una fuente de aire

**Habitación.** El volumen de control que se corresponde con la habitación se modeló empleando una cámara de aire, a la cual, en el puerto térmico se le conectaron los elementos necesarios para simular la transferencia de calor con el exterior de la habitación, así como el calor generado en el interior de la propia habitación (valores estimados de generación de calor generados por el cuadro eléctrico, por el motor de la bomba y por el motor del ventilador).



Imagen 4-25: Componente para simular una cámara de aire con un puerto térmico que admite múltiples entradas

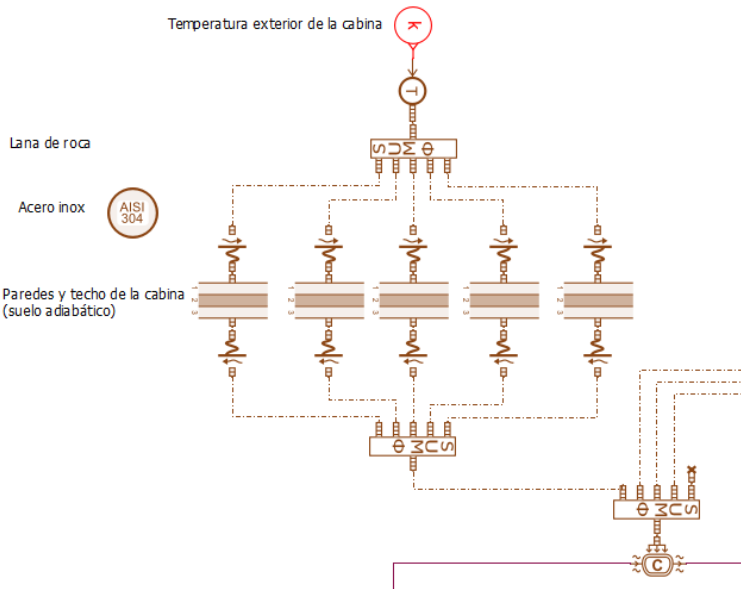


Imagen 4-26: Componentes empleados para simular las paredes de la habitación y la transferencia de calor con el exterior

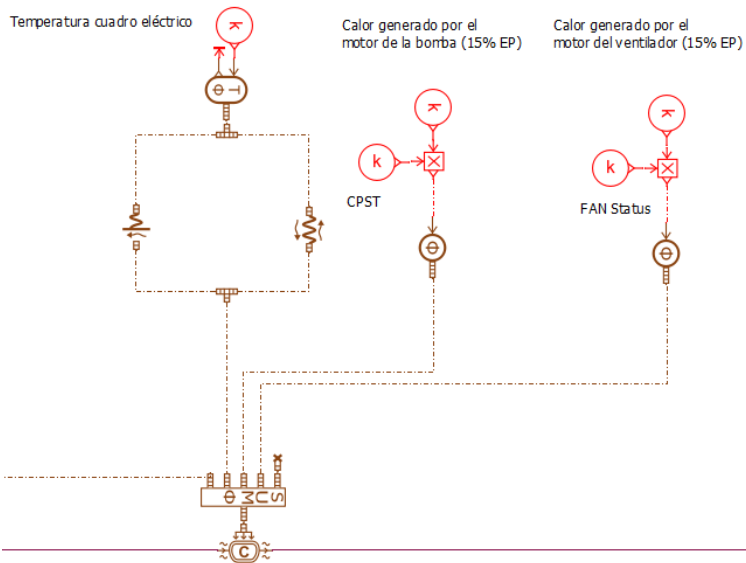


Imagen 4-27: Componentes empleados para simular la transferencia de calor desde el cuadro eléctrico y el calor generado por los motores de la bomba y del ventilador

Como se adelantó en el capítulo 3, en general, en el entorno del gemelo digital se pueden clasificar los modelos de simulación de comportamiento del sistema real en modelos para simulación en tiempo real (modelos que se ejecutan de forma paralela al funcionamiento del sistema que buscan replicar su comportamiento) y modelos de predicción de comportamiento. Aunque los componentes básicos de ambos modelos son, en esencia, iguales, si existen diferencias entre el modelo de predicción de comportamiento y el modelo para simulación en tiempo real.

### Modelo para predicción de comportamiento.

El modelo que permite realizar predicciones de comportamiento del sistema incluye, además de los componentes físicos, el sistema de control del sistema. El control del sistema contiene lo siguiente:

- Criterios de arranque y parada del sistema en modo verano y en modo invierno. Se definieron empleando un diagrama de estados en AMESIM (Siemens-AMESIM, 2016).

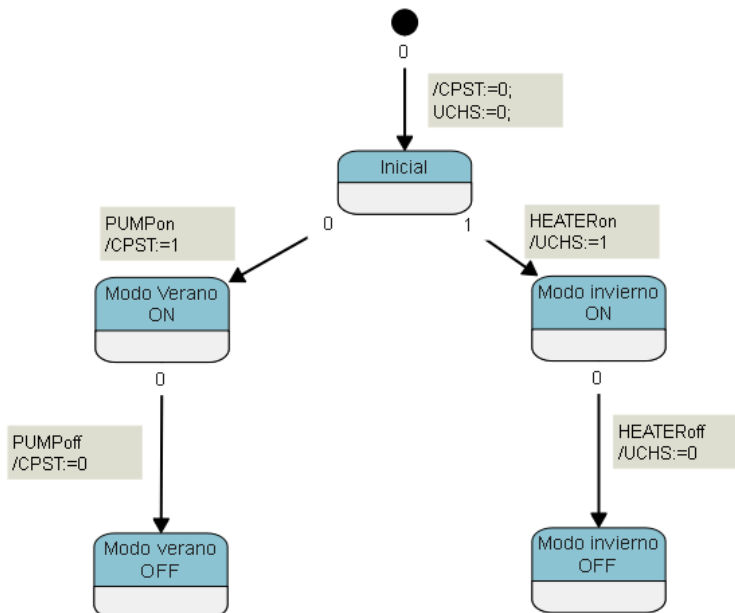


Imagen 4-28: Diagrama de estados definido en AMESIM para simular el control ON/OFF del sistema



Name	Value	Type	Default
<b>Input variables</b>			
AMBT		Real	
UCTSP		Real	
UCOM		Real	
<b>Continuous output variables</b>			
<b>Discrete output variables</b>			
CPST		Real	0
UCHS		Real	0
<b>Parameters</b>			
<b>Events</b>			
PUMPOff	AMBT < UCTSP - 2		
HEATERoff	AMBT > UCTSP + 1.5		
PUMPOn	UCOM = 1		
HEATERon	UCOM = 0		

**Imagen 4-29: Variables y eventos definidos en el diagrama de estados**

Las variables de entrada del diagrama de estados son: la temperatura ambiente, AMBT, la temperatura de consigna, UCTSP y el modo de funcionamiento del sistema (verano/invierno), UCOM. Las variables de salida serán el estado de la bomba y del ventilador (ON, OFF).

En modo verano: Si UCOM = 1, la bomba arrancará, y parará cuando la temperatura ambiente sea menor que la temperatura asignada como temperatura de consigna - 2°C.

En modo invierno: Si UCOM = 0, el calentador arrancará, y parará cuando la temperatura ambiente sea mayor que la temperatura asignada como temperatura de consigna + 2°C.

- Selección de la velocidad del ventilador. La selección de la velocidad del ventilador se llevó a cabo también diseñando un diagrama de estados, trasladando a él la información contenida en la tabla 4-2:

Tabla 4-2: Tabla en la que se definen las condiciones de velocidad del ventilador

	$\Delta H < -15\%$	$-15\% < \Delta H < +5\%$	$+5\% < \Delta H < +20\%$	$\Delta H > +20\%$
$\Delta T < 2^\circ\text{C}$	Media 1085 rpm	Baja 775 rpm	Baja 775 rpm	Baja 775 rpm
$2^\circ\text{C} < \Delta T < 4^\circ\text{C}$	Alta 1395 rpm	Media 1085 rpm	Media 1085 rpm	Baja 775 rpm
$\Delta T > 4^\circ\text{C}$	Alta 1395 rpm	Alta 1395 rpm	Media 1085 rpm	Baja 775 rpm

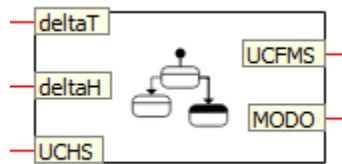


Imagen 4-30: Esquema resumido del diagrama de estados para calcular la velocidad del ventilador con las variables de entrada y salida.

- Cálculo del caudal de agua requerido por el unit cooler, y que es el valor que se envía a la válvula para regular la apertura, según las expresiones aportadas por el suministrador del equipo, y determinadas en base a cálculos experimentales.

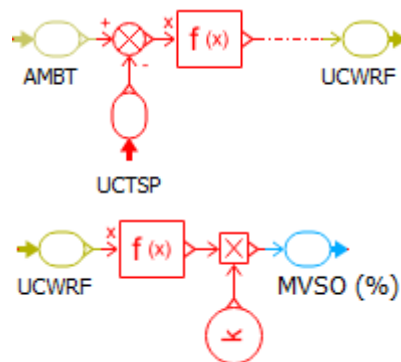


Imagen 4-31: Representación en AMESIM de los componentes creados para definir el control de la válvula.

La expresión para determinar el caudal de agua requerido, en función de la temperatura de la cabina y de la temperatura objetivo

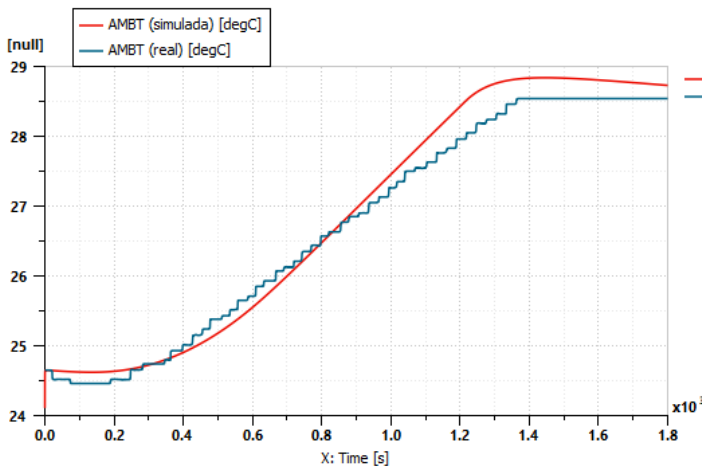
que se busca alcanzar en ella, responde a una expresión polinómica, aportada por el fabricante.

La válvula cuenta con un controlador inteligente que modula la apertura en base al caudal objetivo UCWRF. La expresión que determina la apertura de la válvula, en función de ese caudal (y por lo tanto el input que tendremos que darle a la válvula) se determina a partir de la siguiente expresión:

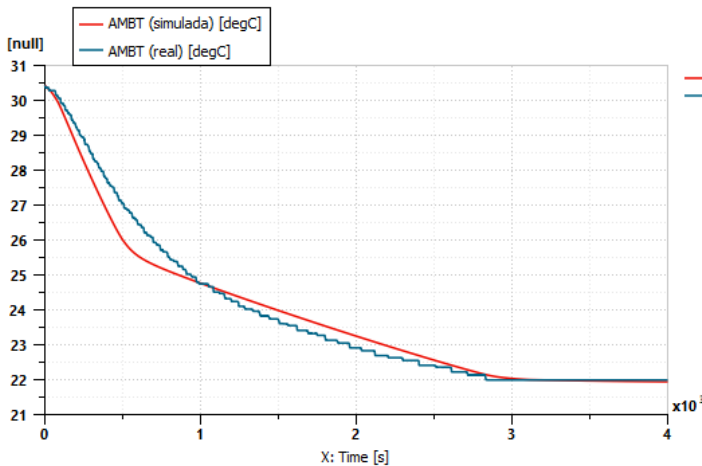
$$MVSO = 0.2766 \cdot UCWRF^3 - 0.4073 \cdot UCWRF^2 + 0.6127 \cdot UCWRF + 0.0062 \quad (4-16)$$

A diferencia de lo que se pudo ver en el caso de OpenModelica, AMESIM no admite código, pero es posible implementar funciones matemáticas mediante el componente  $f(x)$  de la imagen 4-31.

En las siguientes imágenes se pueden ver algunos de los resultados obtenidos en la predicción de comportamiento del sistema (en la imagen 4-32, la evolución de la temperatura en el interior de la cabina con el sistema funcionando en modo invierno, y en la imagen 4-33 la evolución de la temperatura en el interior de la cabina con el sistema trabajando en modo verano):



**Imagen 4-32: Comparativa de los resultados obtenidos para la evolución de la temperatura en la cabina para el modo invierno**



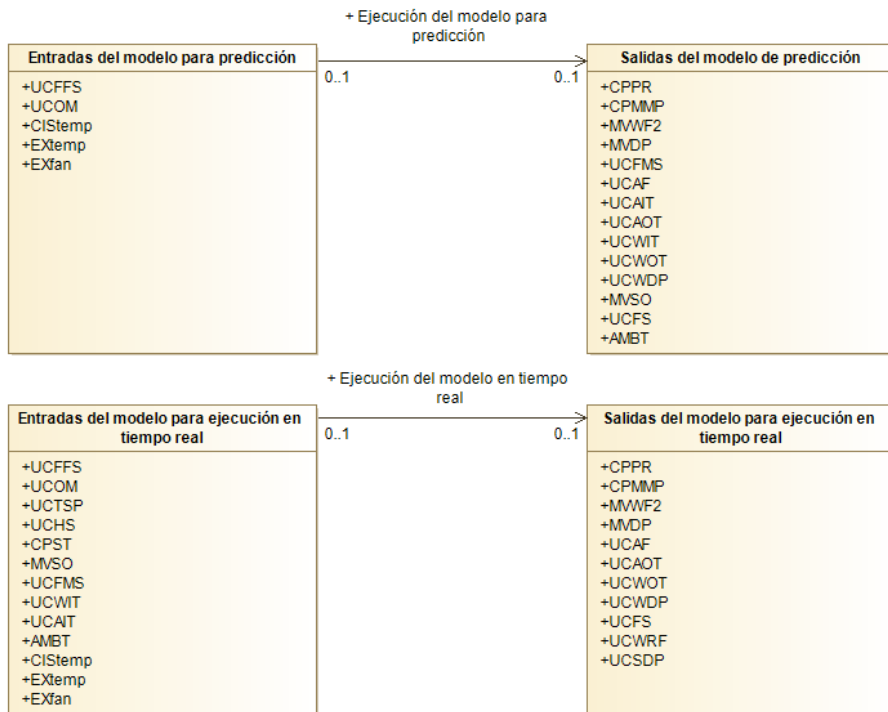
**Imagen 4-33: Comparativa de los resultados obtenidos para la evolución de la temperatura en la cabina para el modo verano**

### Modelo para ejecución en tiempo real

Los modelos para ejecución en tiempo real o modelos “real-time” son, de algún modo, la esencia de los modelos aplicables al gemelo digital: un modelo que es capaz de simular en paralelo a su homólogo físico, a partir de determinadas señales recogidas en él. A partir de ahí, y en cada instante de tiempo de la simulación el modelo actualiza su comportamiento, ofreciendo resultados, que pueden ser comparados con los que ofrecen los sensores presentes en el sistema físico. Los modelos de simulación en tiempo real también pueden ser empleados como sensores virtuales: si disponemos de un modelo de alta fiabilidad sobre el cual se han podido validar sus diferentes salidas, en la práctica, podremos evitar la instalación de sensores que por limitaciones económicas o físicas no puedan ser instalados en el sistema real.

En cuanto a los componentes que contiene el modelo 1D, se puede decir que el modelo real time es, en esencia, igual que el modelo de predicción (y que es un modelo de paso variable), aunque se ha de tener en cuenta lo siguiente: En primer lugar, las entradas y las salidas del modelo real time no serán las mismas que las del modelo para predicción, ya que en el caso del modelo real time hay señales reales del sistema que se leen del sistema físico y sirven de entradas al modelo (en el caso del modelo de predicción todas las señales deben de ser simuladas). Esto se puede ver en la imagen 4-34 en la que

se presentan los diagramas definidos en SysML para definir las entradas y las salidas de los modelos, respecto a las variables definidas en el modelo de datos del apéndice.



**Imagen 4-34: Diagrama de bloques (SysML) para definición de entradas y salidas de los modelos de simulación**

En segundo lugar, dado que el modelo de simulación real time debe de ejecutarse en una plataforma para real time (en este caso en el software LiveTwin de Siemens que se encuentra en un equipo “Nanobox” también de Siemens) es necesario que cumpla unas determinadas condiciones en cuanto al paso (step) que emplea el modelo para ser calculado. AMESIM, determina de forma automática el paso de tiempo óptimo para simular los componentes que forman parte del modelo (paso variable), alguno de estos pasos de tiempo es considerablemente pequeño, y, en general, las plataformas para simulación en tiempo real no admiten esos pasos de tiempo (suele conocerse un valor aproximado del valor mínimo que se puede dar al paso de tiempo, en el caso de Livetwin, el personal de apoyo de Siemens al que se consultó indicó que el mínimo paso de tiempo que podía aceptar Livetwin estaría en torno a 0.5 ms.

Por ello, fue necesario modificar el modelo para poder alcanzar resultados (razonables) empleando ese paso de tiempo. Esta transformación para que el modelo pueda ser real time, requiere de un proceso laborioso y que, en algunos casos, puede llegar a implicar una pérdida de fiabilidad en los resultados: es necesario hacer uso de la herramienta Performance Analyzer, la cual es, en general, una herramienta que permite analizar el proceso de simulación del modelo, de cara a optimizar su cálculo. En él, se pueden analizar múltiples parámetros relativos a la simulación, los cuales pueden ralentizar o incluso impedir la simulación. El tiempo de una simulación vendrá determinado por la complejidad del modelo, pero también por:

- El número y tipo de ecuaciones que se quieren resolver.
- El número de discontinuidades.
- La velocidad a la que varían las variables en el modelo.
- La no linealidad de las ecuaciones.
- Tamaño del paso de tiempo de la integración.
- Velocidad del computador.
- Configuración de software.

El objetivo del analizador de rendimiento es recopilar información sobre el modelo y sobre el proceso de resolución para evaluar cómo se puede mejorar el rendimiento. En lo referente al paso de tiempo del modelo, esta herramienta ofrecerá un análisis de los elementos que requieren de un paso de tiempo más pequeño, así como el paso de tiempo máximo que se puede emplear como paso de tiempo fijo en el modelo. En resumen, lo que habrá que hacer, será, haciendo uso de esta herramienta, modificar los elementos que más afectan al cálculo del paso hasta conseguir un paso de tiempo que se encuentre por encima del valor fijado por la plataforma de ejecución del modelo.

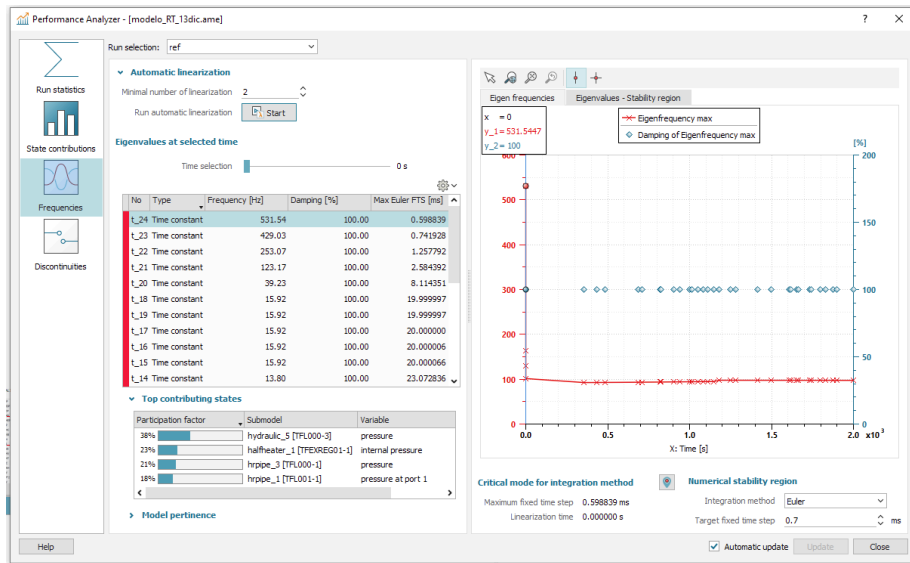


Imagen 4-35: Herramienta "Performance Analyzer"

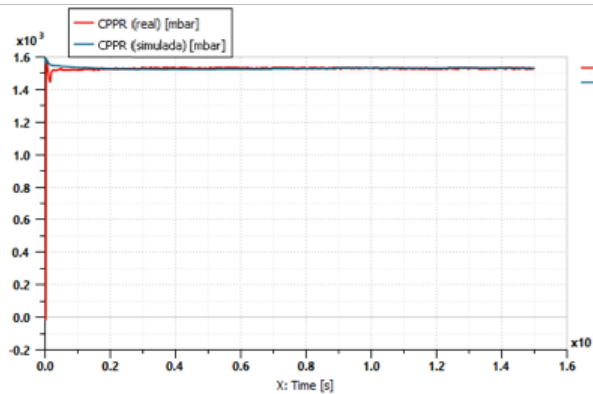
En resumen, un modelo de AMESIM de paso variable, implica que AMESIM determinará el paso óptimo para cada componente. Pero para ejecutar un modelo real time es necesario que ese paso sea fijo, por lo que, si se selecciona el menor de ellos, para poder ejecutarlo será necesaria una capacidad de cálculo muy grande. Por ello, es necesario "bajar" ese paso de cálculo fijo a un valor asumible por la plataforma real-time (en torno a 0.5 - 0.7 ms). Conseguir esto, en la práctica, requiere aumentar el volumen de algunos componentes, lo que, en el caso de un sistema hidráulico, puede significar alteraciones en las pérdidas de carga, por lo que es necesario ir evaluando el modelo cada vez que se hace una modificación para comprobar que los resultados siguen siendo satisfactorios.

Para la validación del modelo para simulación en tiempo real se emplearon datos reales del sistema.

**Tabla 4-3: Casos de funcionamiento real del sistema empleados para la validación del modelo de simulación**

	AMBT	UCOM	UCTSP	FFFS
Caso	Temperatura inicial de la cabina (°C)	Modo de funcionamiento	Temperatura objetivo (°C)	Factor de caudal
1	30.57	Verano	23	100%
2	29.64	Verano	21	100%
3	20.25	Invierno	26	100%
4	22.13	Invierno	27	100%
5	18.7	Invierno	28	100%
6	30.43	Verano	24	100%
7	26.77	Verano	22	100%
8	26.12	Verano	22	100%
9	27.14	Verano	23	100%
10	27.55	Verano	25	140%
11	29.25	Verano	25	120%
12	24.65	Invierno	27	130%
13	22	Invierno	30	130%
14	20.8	Invierno	31	100%
15	19.2	Invierno	25	110%

Se muestra a continuación un ejemplo de los resultados obtenidos en la comparativa entre las variables simuladas frente a las reales:



**Imagen 4-36: Comparativa entre los valores real y simulado para la variable CPPR (incremento de presión en la bomba)**



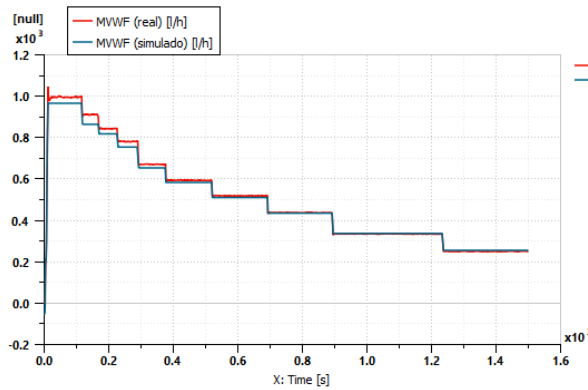


Imagen 4-37: Comparativa entre los valores real y simulado para la variable MVWF (caudal de agua en el circuito hidráulico)

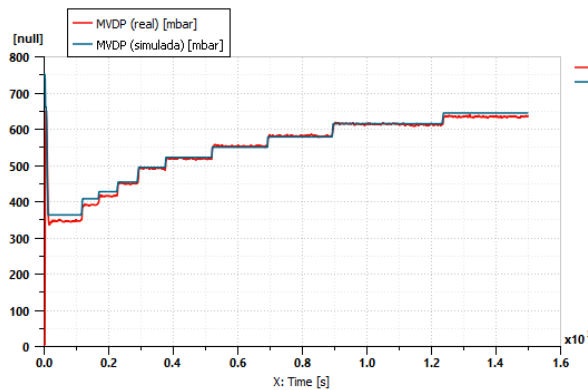


Imagen 4-38: Comparativa entre los valores real y simulado para la variable MVDP (caída de presión en la válvula)

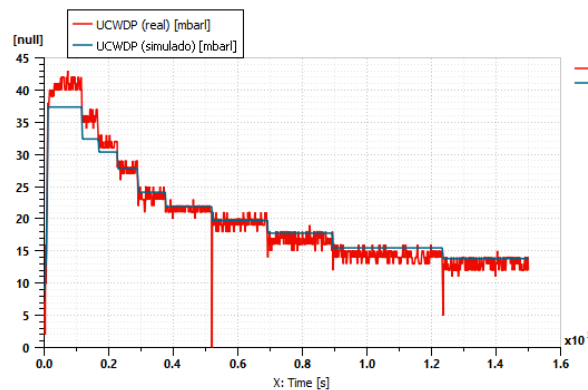


Imagen 4-39: Comparativa entre los valores real y simulado para la variable UCWDP (caída de presión en el serpentín del unit-cooler)

### 4.5.3. Conexión de los modelos de simulación con el gemelo digital.

#### 4.5.3.1. Procesado previo para ejecución de modelos en el entorno del gemelo digital. Exportación a FMU

El primer paso para poder ejecutar los modelos desarrollados en AMESIM en el entorno del gemelo digital, es realizar su exportación a formato FMU. En AMESIM, el proceso de exportación es relativamente sencillo, pero se ha de tener en cuenta que los pasos para realizar la exportación no serán los mismos dependiendo de si el modelo que estamos tratando de exportar es un modelo para predicción o si en cambio es un modelo para su ejecución en tiempo real.

Lo primero que se debe realizar en AMESIM para exportar un modelo al estándar FMI, es preparar un bloque en el que se definen las entradas y salidas que va a tener ese FMU, es decir, qué variables serán necesarias que sean asignadas desde fuera hacia modelo (ya sean señales reales leídas de sensores en el sistema real, o aquellos parámetros de entrada según los cuales queremos que el modelo realice una simulación) así como las salidas debe facilitar el modelo una vez ejecutado. En la imagen 4-40 se muestra el bloque definido para el modelo de predicción de comportamiento del HVAC.

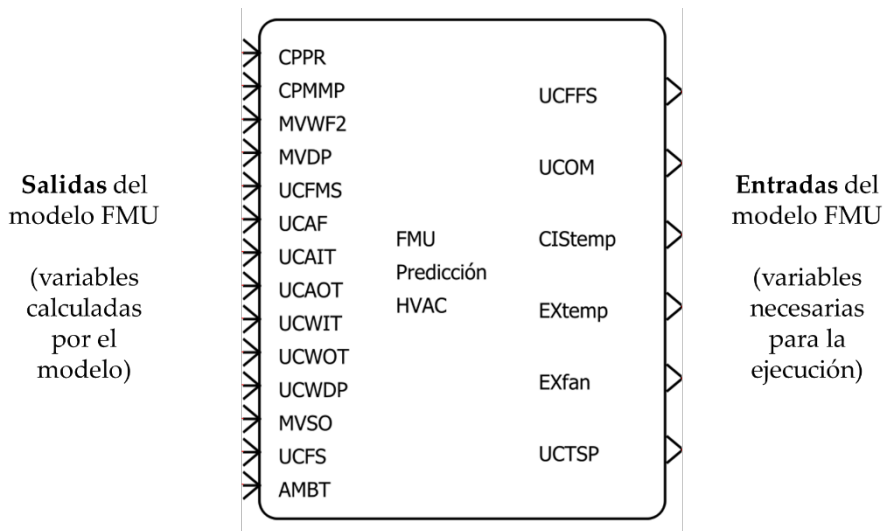


Imagen 4-40: Bloque definido en AMESIM para poder realizar la exportación de un modelo de simulación a formato FMU (modelo para predicción)

Los parámetros de entrada que ha de leer el modelo para ejecutarse son los siguientes:

- UCFFS: Factor caudal. Valor entre 60% y 140%, que multiplica las revoluciones del ventilador (es configurable desde el HMI del sistema HVAC).
- UCOM: Variable que define si el sistema se encuentra en modo verano o modo invierno.
- CISTEMP: Temperatura del espacio exterior a la cabina.
- EXtemp: Temperatura exterior.
- EXfan: Condición de encendido o apagado de un ventilador para renovación del aire en el interior de la cabina. (en este modelo se ha considerado que siempre está apagado).
- UCTSP: Temperatura objetivo que se quiere alcanzar en el interior de la cabina.

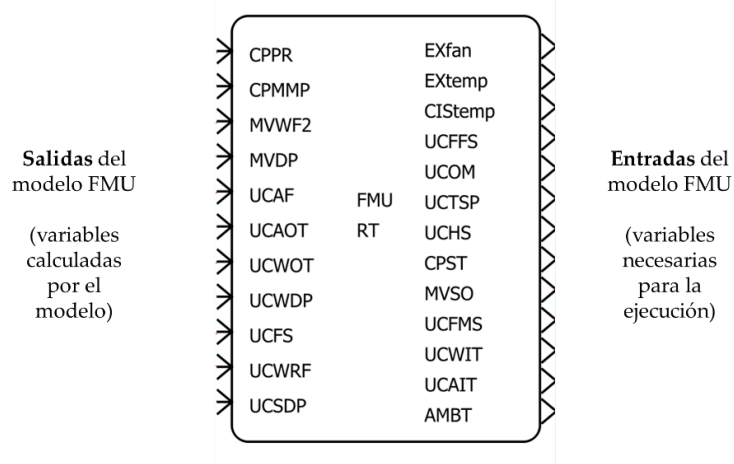
Adicionalmente, el modelo necesitará un parámetro de entrada en el que se indique la temperatura inicial de la cabina, pero en este modelo se trata de un parámetro interno del componente que se ha usado para simular la cabina, por lo que no puede ser definido como un input del FMU. La solución a este inconveniente es hacer visible ese parámetro interno del componente cabina (y que se ha denominado *Tinit*) dentro del modelo AMESIM, de este modo, antes de ejecutar el FMU de forma externa a AMESIM será posible asignar un valor a ese parámetro. El inconveniente que presenta esto es que si bien, el resto de los parámetros pueden ser introducidos de forma manual a través de una ventana gráfica de WinccOA o leídos de los sensores del sistema real, en el caso de esa variable *Tinit*, es necesario acceder a Livetwin (la plataforma en la que se ejecutan los modelos de simulación, y modificar ese valor). Este tema se consultó con el personal que dio soporte por parte de Siemens a los desarrollos de modelos de AMESIM, pero no ofrecieron ninguna solución alternativa.

Por otro lado, las variables de salida del modelo FMU serán:

- CPPR: Incremento de presión en la bomba.
- CPMMP: Potencia mecánica generada por la bomba.
- MVWF2: Caudal de agua.
- MVDP: Pérdida de presión en la válvula.
- UCFMS: Velocidad de giro del motor del ventilador.

- UCAF: Caudal de aire suministrado por el ventilador.
- UCAIT: Temperatura de entrada de aire en el unit cooler.
- UCAOT: Temperatura de salida de aire en el unit cooler.
- UCWIT: Temperatura de entrada de agua en el unit cooler.
- UCWOT: Temperatura de salida de agua en el unit cooler.
- UCWDP: Pérdida de presión de agua en el serpentín del unit cooler.
- MVSO: Porcentaje de apertura de la válvula.
- UCFS: Velocidad de giro del ventilador.

En el caso del modelo para ejecución en tiempo real (partiendo del supuesto que ya se han hecho las transformaciones necesarias para que pueda ser ejecutado en una plataforma real time), el modo de proceder es idéntico al caso anterior. Se ha de definir el bloque para el interfaz FMU, en el que se establecen las entradas y las salidas que va a tener el modelo. En este caso, como se puede ver en la imagen 4-41, el número de entradas es mucho mayor que en el caso de predicción, ya que en este modelo ejecutándose en tiempo real, se tratará de alimentarlo con determinadas señales reales de entrada, a partir de lo cual tendremos diferentes salidas (señales simuladas), de las cuales, algunas serán sensores virtuales (ya que no existe una señal real procedente de un sensor), mientras que otras serán variables simuladas que se emplearán para comparar en tiempo real el funcionamiento del sistema físico con el esperado según el modelo, de cara a detectar fallos o problemas de funcionamiento.



**Imagen 4-41: Bloque definido en AMESIM para poder realizar la exportación de un modelo de simulación a formato FMU (modelo para tiempo real)**

Las señales de entrada del modelo, para cada instante en el que se ejecute la simulación, serán:

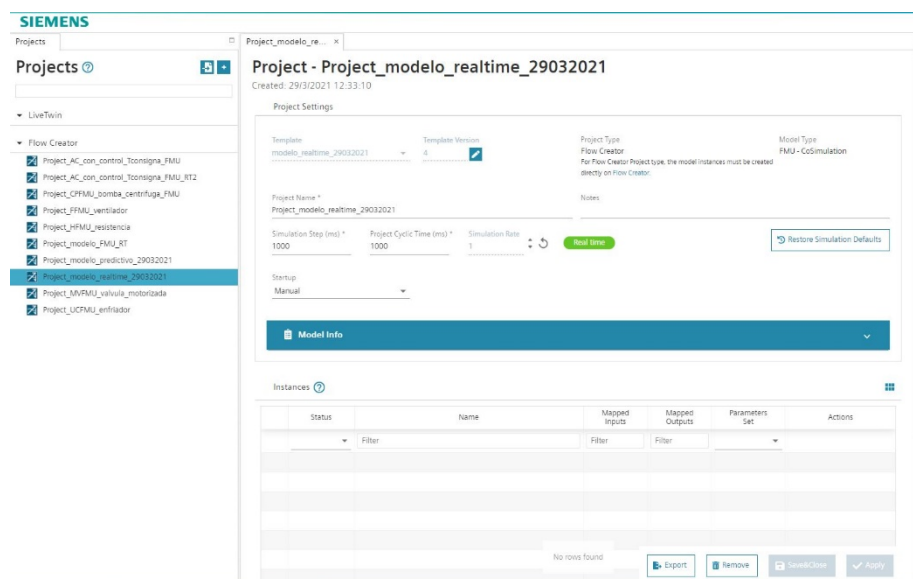
- EXtemp: Temperatura exterior.
- EXfan: Condición de encendido o apagado de un ventilador para renovación del aire en el interior de la cabina. (en este modelo se ha considerado que siempre está apagado) .
- CISTEMP: Temperatura del espacio exterior a la cabina.
- UCFFS: Factor caudal. Valor entre 60% y 140%, que multiplica las revoluciones del ventilador (es configurable desde el HMI del sistema HVAC).
- UCOM: Variable que define si el sistema se encuentra en modo verano o modo invierno.
- UCTSP: Temperatura objetivo que se quiere alcanzar en el interior de la cabina.
- UCHS: Status del calentador (ON/OFF).
- CPST: Status de la bomba (ON/OFF).
- MVSO: Porcentaje de apertura de la válvula.
- UCFMS: Velocidad de giro del motor del ventilador.
- UCWIT: Temperatura de entrada del agua en el unit cooler.
- UCAIT: Temperatura de entrada del aire en el unit cooler.
- AMBT: Temperatura del aire en el interior de la cabina.

Y las variables que devolverá el modelo en cada ejecución:

- CPPR: Incremento de presión en la bomba.
- CPMMP: Potencia mecánica generada por la bomba.
- MVWF2: Caudal de agua.
- MVDP: Pérdida de presión en la válvula.
- UCAF: Caudal de aire suministrado por el ventilador.
- UCAOT: Temperatura de salida de aire en el unit cooler.
- UCWOT: Temperatura de salida de agua en el unit cooler.
- UCWDP: Pérdida de presión de agua en el serpentín del unit cooler.
- UCFS: Velocidad de giro del ventilador.
- UCWRF: Caudal de agua requerido por el unit cooler.
- UCSDP: Pérdida de presión del aire en el filtro de entrada del unit cooler.

#### 4.5.3.2. Ejecución de modelos en el entorno del gemelo digital

En este caso, el PC que da el servicio en la capa Edge será el que se encargue de ejecutar los modelos FMU. En el caso de uso propuesto, se empleó la Plataforma LiveTwin de Siemens (Siemens, 2020b, 2020a). LiveTwin es una aplicación basada en navegador que se utiliza en el nivel edge. Su uso está ligado a la plataforma Siemens Industrial Edge y, en resumen, permite ejecutar fácil y directamente modelos de simulación 1D. En teoría, dicha plataforma podría admitir modelos de AMESIM, aunque lo normal es llevar a cabo la ejecución de modelos en la plataforma LiveTwin exportados a formato FMU. LiveTwin también permite la ejecución o co-simulación de modelos exportados a formato FMU que hayan sido desarrollados en otra plataforma (OpenModelica, Matlab Simulink, etc).

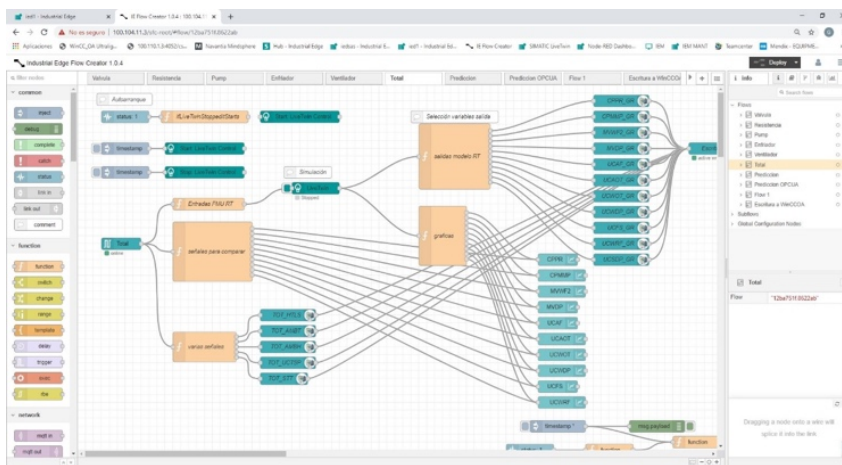


**Imagen 4-42: Visualización de interfaz Industrial Edge de Siemens: LiveTwin - Información de modelo FMU**

En el caso de los modelos real time, como ya se ha comentado en este documento, se requiere de un pre- procesamiento del modelo para fijar un paso de cálculo fijo que sea asumible por la plataforma en la que se vayan a ejecutar. Se trata de un requisito que han de cumplir los modelos de simulación para poder ser ejecutados en lo que se denominan plataformas para Real Time como es el caso de LiveTwin, o como podrían ser las plataformas: Concurrent (Hexagon, 2019), dSpace (DSpace, 2021) o ETAS Labcar (ETAS, 2016).

En el caso de los modelos de paso variable, con llevar a cabo la exportación a FMU es suficiente, si bien, se debe de tener en cuenta que la exportación a FMU se debe de realizar en el mismo sistema operativo en el que se vaya a ejecutar en el entorno del gemelo digital.

Hasta ahora se ha descrito solo la parte de la carga en el edge de los modelos de simulación, pero el desarrollo necesario para la ejecución de los modelos se realiza al completo, también en el entorno de edge, pero a través de una herramienta denominada Flowcreator, mediante la que se pueden realizar las diferentes conexiones de entrada y salida de los modelos, de forma gráfica (con poca necesidad de programación) como se puede ver en la imagen 4-43. Desde allí se conectan las señales de los sensores que han de servir como inputs a la simulación (a través de la conexión con la plataforma SCADA de Siemens, WinCC OA) se establecen los modelos que se quieren simular (si existe o no co-simulación) y se establecen las salidas de los modelos, las cuales, en general, pasarán también a WinCC OA para su visualización, ya sea: visualización de comportamiento esperado según simulación e modelos de predicción, visualización de sensores virtuales, comparación de valores reales frente a simulados, cálculo de alarmas, etc.

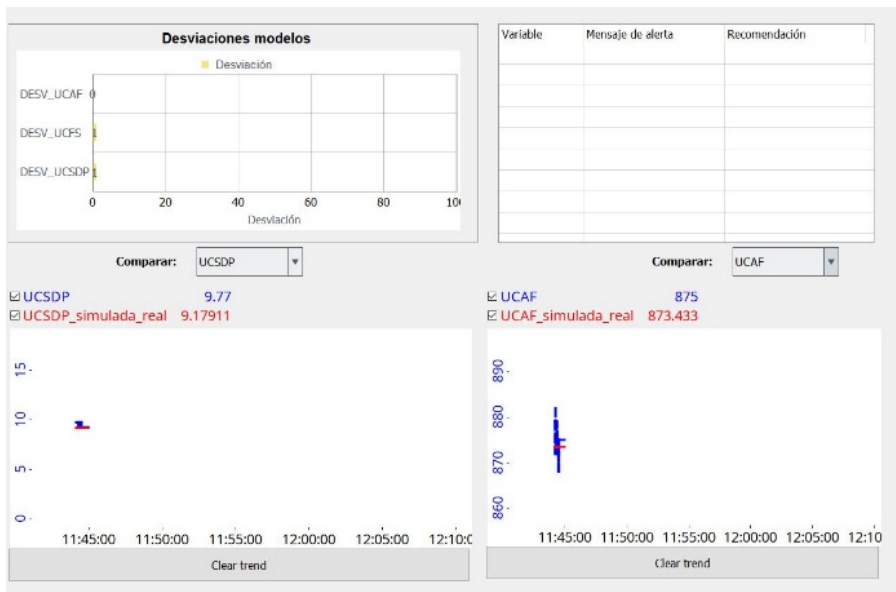


**Imagen 4-43: Ventana de Industrial Edge de Siemens: Flowcreator**

En el caso de que el foco de interés sea realizar una comparativa entre señales reales y simuladas, en Wincc OA, además de representar en gráficas ambas señales, se procede a calcular el porcentaje de error entre la variable real y



la simulada, y se establecen rangos, a partir de los cuales se considera que el sistema real puede estar fallando. Si esto sucede, se muestra una ventana de alerta, y un mensaje de recomendación, por ejemplo: en el caso del ensuciamiento del filtro, se comparan las señales correspondientes a la diferencia de presión en el filtro de entrada del aire y el caudal de aire en el unit cooler. Si la diferencia de presión en el filtro varía más de un 20% entre el valor real y el simulado, salta un mensaje de alerta, en el que se le indica al operario que revise el filtro de aire y considere la posibilidad de que sea necesario reemplazarlo.



**Imagen 4-44: Visualización de señales reales frente a señales simuladas, indicadores de error entre valores reales y simulados y ventana de alertas (sobre WinCC OA)**

#### 4.5.4. Caso de uso. Detección de fallos en tiempo real

Como aplicación práctica del desarrollo de los modelos de simulación del sistema y de su ejecución en el entorno del gemelo digital, se propuso realizar una prueba de concepto para emplear el gemelo digital en la detección de fallos en el sistema en tiempo real. Para la realización de las primeras pruebas, se seleccionó la detección de la obstrucción del filtro de entrada de aire en el unit cooler. En primer lugar, se identificaron las variables que se verían afectadas en un supuesto fallo por obstrucción en el filtro, y que serían: en primer

lugar, la diferencia de presión en el propio filtro. En segundo lugar, el caudal de aire. Para la realización de estas pruebas lo que se realizó fue colocar una sobre-capa sobre el filtro, fabricada con el mismo material que este, tratando de conseguir un efecto similar al que se pueda producir por un filtro sucio.

Se programó una visualización en la que se pudiera ver:

- La comparativa de la señal real con la señal calculada.
- Una tabla en la que se indique la desviación entre las señales.
- Una ventana de alerta cuando la desviación entre las señales real y calculada supere el umbral fijado.

Se establecieron 2 rangos de desviación de las variables medidas respecto a las obtenidas en el modelo:

- Alarma media si la diferencia entre la señal medida en el sistema y la señal simulada difería en más de un 10%.
- Alarma alta si la diferencia entre la señal medida en el sistema y la señal simulada difería en más de un 30%.

En las siguientes imágenes se puede ver la secuencia de realización de la prueba:

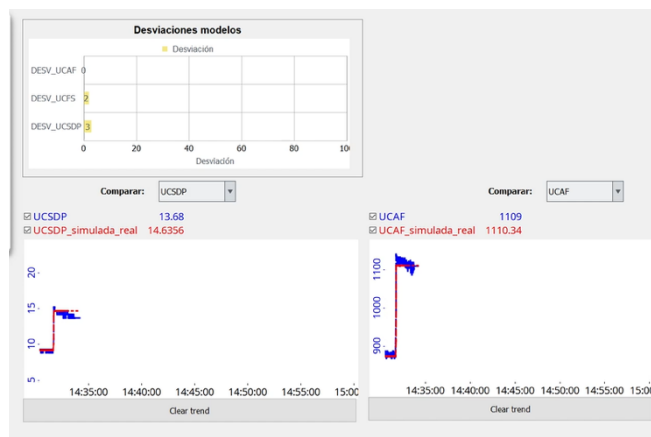


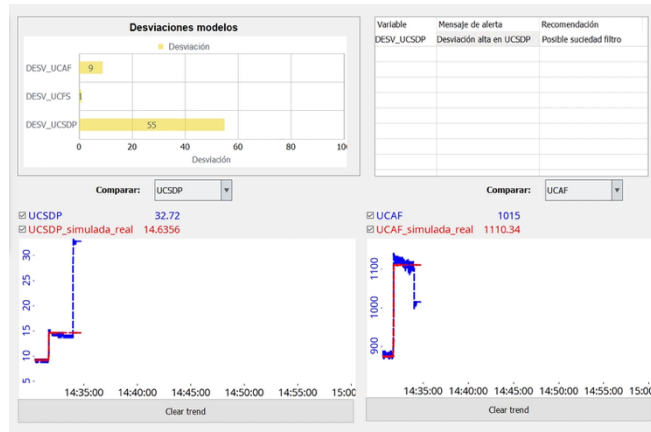
Imagen 4-45: Pruebas de detección de fallos (1/3)

En la imagen 4-45, se muestra el sistema en funcionamiento normal. Las variables de interés que se están comparando son:

- UCSDP: Caída de presión en el filtro de entrada del aire.
- UCAF: Caudal de aire.

El incremento que se produce en las dos variables es debido a que se aumentan las revoluciones del ventilador y, en consecuencia, el caudal de aire, y la diferencia de presión en el filtro de entrada de aire. Pero se ve que se comportan de igual forma el modelo y las señales reales.

A continuación, se obstruye el filtro de entrada de aire:



**Imagen 4-46: Pruebas de detección de fallos (2/3)**

Tal y como se observa en la imagen 4-46, al obstruir el filtro de entrada de aire, aumenta la diferencia de presión en el filtro y además el caudal de aire cae (señales reales). Las señales simuladas se mantienen. Salta la alarma de alta de diferencia de presión en el filtro, ya que la diferencia entre real y simulada pasa del 30%. El caudal de aire cae menos de un 10%, por lo que no salta ninguna alarma, si bien en las gráficas se puede observar que hay una pequeña caída en caudal.

Finalmente, como se muestra en la imagen 4-47, al retirar de nuevo la obstrucción del filtro, ambas señales de caudal y diferencia de presión (real y simulada) vuelven a converger.

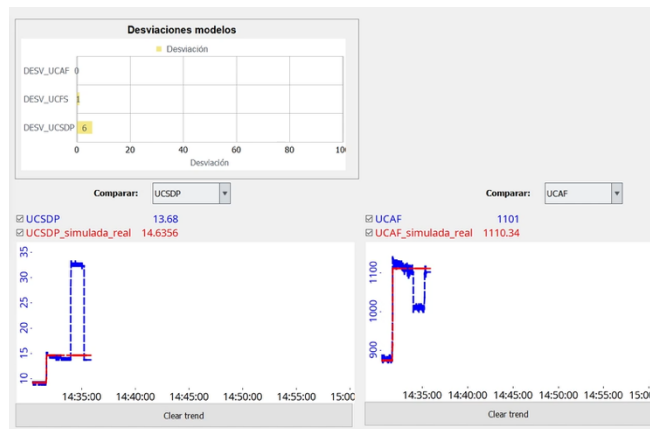


Imagen 4-47: Pruebas de detección de fallos (3/3)

#### 4.5.5. Conclusiones

Se ha analizado un segundo despliegue para implementación del gemelo digital de un sistema HVAC, basado en una arquitectura no centralizada en una plataforma de IoT, en la que parte de las funcionalidades del gemelo digital se encuentran en la capa de computación Edge (en especial, los modelos de simulación o de machine learning).

La mayor parte del trabajo realizado se ha centrado en aspectos relacionados con el desarrollo de modelos de simulación de comportamiento del sistema y con su ejecución en el entorno del gemelo digital:

- Se han desarrollado en la herramienta para modelado de sistemas, AMESIM, modelos de simulación para predicción de comportamiento y para ejecución en tiempo real del sistema HVAC. Al igual que en el despliegue centrado en la plataforma de IoT, la idea fue exportar los modelos desarrollados a modelos de tipo FMU, que serán los que se ejecuten en el entorno del gemelo digital.
- Obtener un modelo de simulación que posea capacidad de ejecución en tiempo real en una plataforma de Edge computing, como Edge Livetwin de Siemens, requiere que el modelo original de AMESIM sea procesado para que pueda ser ejecutado empleando un paso de tiempo determinado por la plataforma

en la que se va a ejecutar. Se procesó el modelo original para su ejecución en tiempo real.

- Ambos modelos fueron ejecutados en el entorno Edge Livetwin. Los modelos han de ser exportados al estándar FMU desde el mismo sistema operativo en el que van a ser ejecutados, en este caso, SO Linux.
- Se validó el uso del modelo de simulación en tiempo real para la detección de fallos, en el caso propuesto, el ensuciamiento del filtro de entrada de aire en el unit cooler. Para ello, además de la ejecución de los modelos en Edge Livetwin, se desarrolló la visualización de los resultados en WinCC OA. La visualización contenía una comparativa entre valores reales y simulados de las variables de interés para el fallo, así como el cálculo de porcentaje de error entre la variable real y la simulada. Además, si ese porcentaje se encontraba por encima de un umbral, se hacía saltar una alarma (de nivel medio o de nivel alto, en función del error entre los valores real y simulado) con recomendaciones para resolver el fallo en el sistema.

Por otro lado, como en esta etapa no fue posible validar el uso de modelos de machine learning para la predicción de comportamiento o para la detección y predicción de fallos, se mantiene como posible trabajo futuro lo siguiente:

- Se dispone de una importante cantidad de datos históricos de funcionamiento, así como un modelo CFD del unit cooler que pueden ser empleados para desarrollar un modelo de orden reducido (ROM). Este modelo puede ser planteado como un desarrollo propio o empleando la herramienta ROM Builder que incluye AMESIM.
- Dentro de los datos históricos de los que se dispone, se incluyen los datos recogidos para dos fallos inducidos en el sistema: el fallo que se empleó en el caso de uso (obstrucción del filtro de aire) y otro fallo que fue inducido en el sistema, en el que se aflojó la correa que conecta el motor del ventilador del unit cooler con el propio ventilador. Se tratará de usar estos datos para validar el uso de algoritmos basados en IA para la detección de estos fallos en el sistema.

Debido a que en este planteamiento no se trabajó en el desarrollo de todas las funcionalidades, no es posible realizar un análisis completo de ventajas e inconvenientes, sin embargo, si se pueden hacer las siguientes consideraciones sobre el uso de edge computing. Como principales ventajas se ha observado:

- Reducción de tiempos de respuesta y latencia, lo cual puede ser un aspecto relevante especialmente en el caso de sistemas críticos del buque.
- Reducción de ancho de banda para traspaso de información a capas superiores, ya que se evita el envío de información sin procesar.

Las principales desventajas identificadas se pueden resumir en los siguientes aspectos:

- Se requiere disponer de mayor capacidad de almacenamiento en la capa edge.
- Existe mayor riesgo de pérdida de datos importantes.
- El coste se ve incrementado, ya que en general, los equipos para edge son más costosos que equipos convencionales.

#### **4.6. ANÁLISIS DE COMPATIBILIDAD E INTEGRACIÓN DE MODELOS DE SIMULACIÓN FMU**

Se puede considerar un buque como un gran sistema complejo, o como un agregado de sistemas heterogéneos complejos. La implantación de gemelos digitales desde el punto de vista de implantación de modelos de simulación de los diferentes sistemas que componen el buque pasa por la integración de modelos, que en general, procederán de diferentes herramientas, estarán elaborados por diferentes equipos de trabajo, desde puntos de vista diversos y con diferentes objetivos. En este marco de trabajo, cobra especial relevancia la co-simulación de modelos. En co-simulación, se modelan de forma separada diferentes modelos que después se componen dentro de una simulación global donde cada modelo se ejecuta de forma independiente, compartiendo información en intervalos de tiempo discretos.

Existen trabajos relacionados con la integración de modelos complejos, entre los que destacan los avances de la plataforma OSP, de la que ya se ha

hablado en esta tesis doctoral, y que centran su trabajo en el desarrollo de una plataforma para integración de modelos de simulación y la co-simulación de modelos en buques, basada en el uso del estándar FMI (L. I. Hatledal et al., 2020; Lars Ivar Hatledal, 2021).

Por ello, y como complemento a los estudios realizados con los modelos de simulación (tanto en Openmodelica como en AMESIM) y a su ejecución y presentación en los diferentes entornos considerados para el gemelo digital (plataforma de IoT Thingworx o Edge Livetwin, WINCC OA) se decidió realizar un estudio de compatibilidad e integración de modelos de simulación en formato FMU: por un lado, se evaluó la posibilidad de importar y simular en las diferentes herramientas de simulación disponibles, modelos FMU generados desde estas mismas herramientas de simulación, a fin de comprobar la existencia o no de incompatibilidades, de errores en la simulación o de, por ejemplo, incremento en los tiempos de simulación al emplear una herramienta diferente a la nativa. Mientras que, por otro lado, se validaron las herramientas existentes para integración y co-simulación de modelos FMU.

#### **4.6.1. Análisis de compatibilidad de modelos FMU**

En primer lugar, se buscó analizar las posibilidades de intercambio de modelos y de cosimulación de FMUs en diferentes entornos de simulación. Se plantearon combinaciones de modelos FMU generados (en general, modelos de tipo co-simulation, pero también modelos model exchange) en diferentes herramientas: OpenModelica, Simulink, Dymola y ANSYS y se ejecutaron también en las siguientes herramientas o plataformas de simulación: AMESIM, SIMULINK, Librerías de PYTHON para ejecución de FMUs y plataforma de Edge Computing, Livetwin. Excepto en el caso de las librerías de Python, el resto de los entornos de simulación contienen interfaces gráficas, que permiten manejar de forma intuitiva los modelos FMU. En general, se visualizará un FMU como una caja opaca de la que solo se pueden ver los conectores de entrada/salida que se deben conectar con los elementos que corresponda dentro de cada entorno. Se pueden observar los ejemplos analizados en las imágenes 4-48, 4-49 y 4-50.

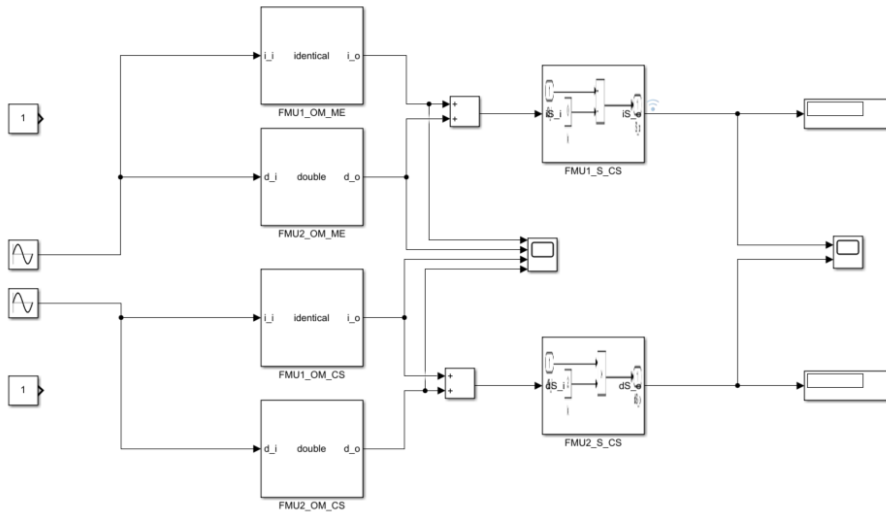


Imagen 4-48: Co-simulación de FMU en SIMULINK

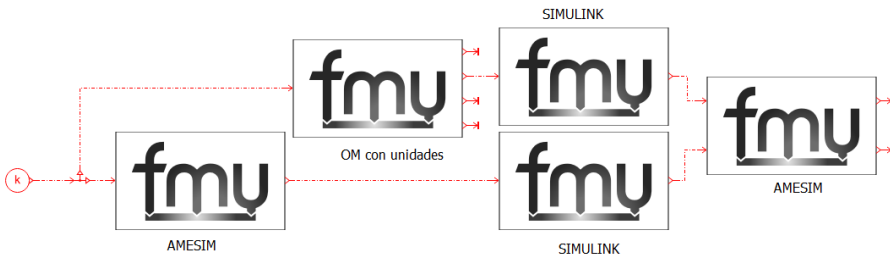


Imagen 4-4-49: Co-simulación de FMU en AMESIM

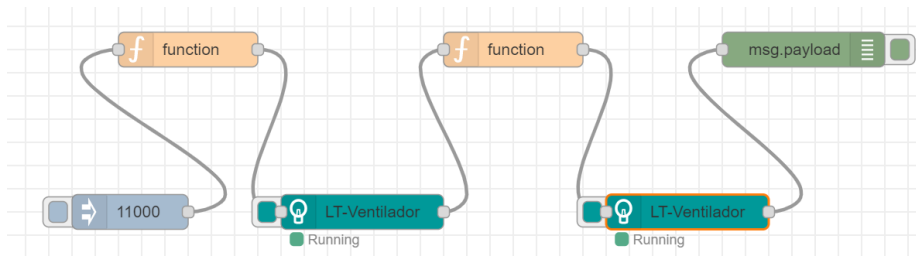


Imagen 4-50: Co-simulación de FMU en Edge LiveTwin

Las principales conclusiones del estudio se pueden resumir en lo siguiente:

- Se ha probado a exportar modelos de simulación a formato FMU desarrollados en OpenModelica, Simulink y AMESIM. Además,



se ha utilizado un FMU de ANSYS, para realizar estas pruebas. Se han importado estos FMU en las plataformas: OpenModelica, Simulink y AMESIM. Además, se han importado en entornos de simulación: Siemens Edge Livetwin, y librería FMPy (desarrollada por Dassault). Se han co-simulado los modelos importados en las diferentes herramientas/entornos.

- Es conocida una limitación en la compatibilidad de FMU. En algunos casos se requiere que el modelo FMU sea exportado en el mismo sistema operativo que el sistema operativo en el que se vayan a ejecutar esos modelos. En general, los modelos FMU que han sido compilados en Linux se ejecutan solamente en entornos de simulación instalados en Linux, a excepción de los generados en OpenModelica. Esto es debido a que permite la opción de generar un identificador único global, que es la equivalencia a la huella de la FMU y los binarios de ambos sistemas operativos (Linux 32 y 64 bits y Windows 32 y 64 bits).
- Es importante tener en cuenta la arquitectura del sistema operativo, si son 32 o 64 bits. Ya que, si los FMUs se van a ejecutar en una máquina de 32 bits, necesitarán de binarios de 32 bits y lo mismo con los 64 bits.
- Las FMUs procedentes de Simcenter Amesim necesitan de licencia para su ejecución, excepto los modelos FMUs para ejecución en tiempo real y aquellos que han sido exportados con la opción "FMU not requiring runtime licenses", opción que no está disponible para todas las licencias y que, además, requiere establecer un paso de tiempo fijo para la exportación del modelo.
- En general, los tiempos de cálculo no se ven gravemente penalizados al emplear modelos FMU importados en herramientas diferentes de la nativa del modelo. Sin embargo, si pueden existir problemas, relacionados con la orquestación de los diferentes modelos, principalmente con la definición de unidades.

#### 4.6.2. Integración de modelos FMU

El análisis realizado en la primera parte se limitó a la co-simulación de FMU dentro de las propias herramientas de simulación, así como en los entornos de simulación que se emplearon en las dos alternativas para despliegue

de gemelo digital (plataforma de IoT y edge computing). Pero existen desarrollos específicos para la integración, orquestación y jerarquización de modelos FMU. Los más importantes son: el estándar SSP, System Structure and Parametrization, que busca definir la estructura, conexiones y parametrización del sistema completo que se quiere simular. Existen diferentes herramientas que han implementado el estándar SSP, una de ellas, OpenModelica. Por otro lado, otro de los desarrollos más avanzados es el que se está llevando a cabo por la plataforma OSP y sus socios DNV-GL, SINTEF Ocean, Kongsberg Maritime y NTNU, que han llegado a poner en disposición del sector una herramienta que permite llevar a cabo la co-simulación de modelos.

De modo que en esta parte del trabajo se optó por analizar el uso del estándar SSP a través de OpenModelica, la herramienta Kopl de OSP y en menor medida el FMU container, del que se hablará más adelante, pero emplea librerías de Python para la integración de FMU.

Para realizar este análisis, se planteó como ejemplo dividir el modelo de simulación del sistema HVAC, en varios modelos para ser interconectados (las salidas de unos componentes serán las entradas de otros). Por lo que se exportaron 9 modelos en formato FMU para su co-simulación: 5 modelos correspondientes a los componentes físicos del sistema (bomba, válvula, ventilador, intercambiador y calentador), un modelo que simula el sistema de control, otro modelo con datos reales de funcionamiento (para emplear como entradas), un modelo para la especificación de los parámetros de entrada de la simulación y uno con las señales de salida agrupadas en un mismo modelo, tal y como se puede ver en la imagen 4-51.

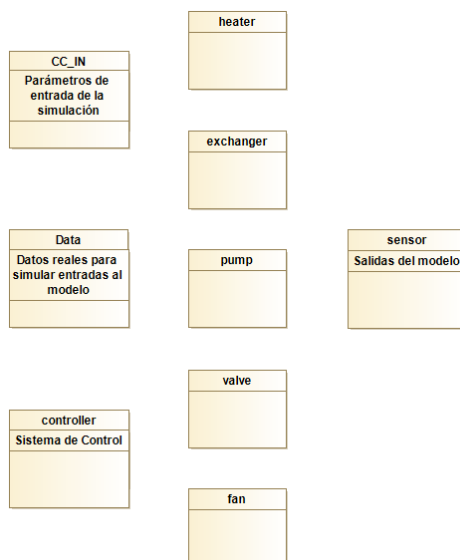


Imagen 4-51: FMUs desarrollados para emplear en las pruebas con SSP y OSP

#### 4.6.2.1. Estándar System Structure and Parametrization (SSP)

El estándar SSP ha sido desarrollado como una extensión al FMI, que permite conectar los modelos FMU entre sí. El SSP ofrece la posibilidad de empaquetar FMUs y sus parámetros, de forma que quedan preparados para ser ejecutados. Además, se puede emplear para archivar simulaciones (.mat) y evitar la pérdida de información. El SSP permite renombrar los conectores a/desde FMUs. Esto es útil cuando se trata de FMUs procedentes de diferentes fuentes, ya que cada uno emplea una nomenclatura distinta y unidades diferentes.

Tabla 4-4: Herramientas que implementan el estándar SSP como complemento al FMI

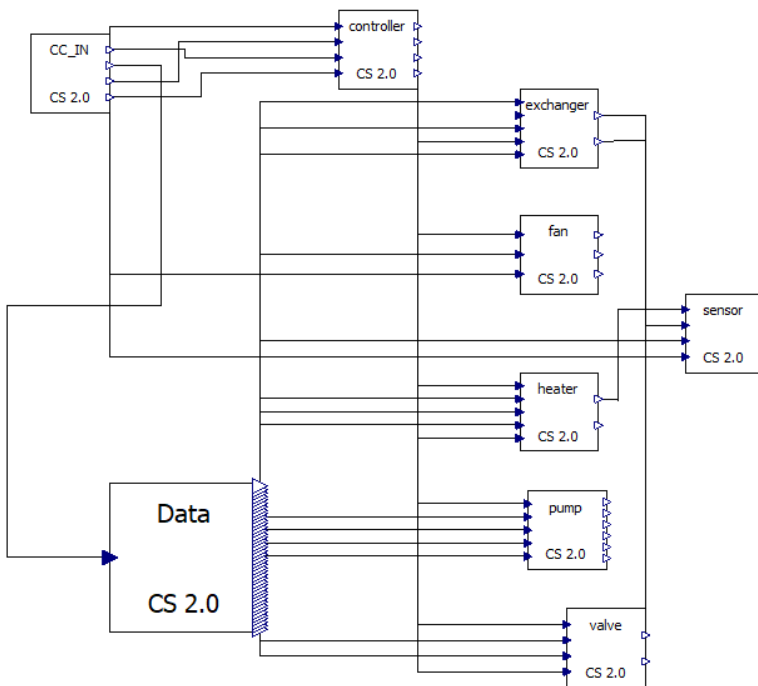
Nombre	Desarrollador
SYNECT Model Management	dSPACE
OMSimulator	OpenModelica
Model.CONNECT	AVL
FMI Bench	PMSF
easySSP	eXXcellent solutions
Simcenter System Architect 2021.1	Siemens

Se ha probado el **OMSimulator** (incorporado en la herramienta OpenModelica) para la ejecución de FMUs bajo el estándar SSP (Mengist et al., 2015;

Ochel et al., 2019; OpenModelica.org, 2019a, 2019b). El procedimiento de trabajo para emplear el entorno OMSimulator es el siguiente:

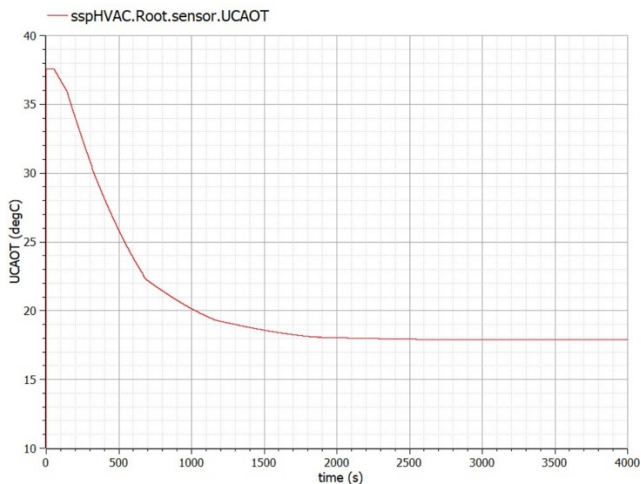
- Generación de un nuevo modelo SSP.
- Especificación del tipo de FMUs a acoplar.
- Añadir sub-modelos (.fmu, .mat y/o .csv ).
- Añadir conexiones.
- Guardar el modelo SSP (y ejecutar, si procede).

Se obtiene un esquema como el que se muestra en la imagen 4-52.



**Imagen 4-52: Orquestación de FMU en OMSimulator**

Los resultados que se obtienen en la simulación se presentan de la misma forma que los de las simulaciones realizadas directamente con OpenModelica, además de la visualización de resultados en forma de gráficas, permite exportar los resultados a un fichero csv, seleccionando las variables que se quieren exportar.



**Imagen 4-53: Gráfica de salida de resultados de co-simulación en la plataforma OMSimulator**

También se testó la orquestación de FMUs para co-simulación en la herramienta **easy-SSP**, (Orchideo, 2019). Una vez generado el SSP, se obtiene un fichero comprimido de tipo zip con extensión. *ssp*, en el cual se encuentra una carpeta con los sub-modelos (FMUs, *.csv* *.mat*), llamada “resources” y un archivo de descripción de la estructura del sistema “SSD” (SystemStructureDescription). Sin embargo, la versión testeada no permite ejecutar la co-simulación, por lo que no ha sido posible valorar la ejecución de modelos.

Se han analizado los siguientes aspectos sobre el uso de una herramienta basada en el estándar SSP para integración de FMU:

- Jerarquía (dependencia entre los conectores de las FMUs): La jerarquía de los FMUs integrados en el SSP es muy clara y viene bien descrita en el archivo XML en el que se define la estructura de la integración (SSD).
- Actualización de modelos (cambiar un FMU antiguo por uno nuevo): Se conectan las entradas y/o salidas del nuevo modelo con los ya existentes, es decir, es suficiente con sustituir el antiguo por el nuevo.
- Gestión de conectores (entradas/salidas de los diferentes FMUs conectados): Se lleva a cabo mediante conectores directos (con ello se define la jerarquía).

- Gestión de parámetros: La parametrización de variables internas se gestiona en los modelos FMUs individuales.
- Gestión de unidades: Se genera un archivo XML (SSV) específico para la gestión de las unidades.
- Herramientas de integración: Hasta la actualidad hay cuatro softwares que los genera y solo dos son libres (OMSimulator y easySSP).
- Entornos de simulación: Solo se ejecutan los SSP en OMSimulator que han sido generados en este.
- Actualizaciones de la herramienta: No hay actualizaciones nuevas de SSP desde el lanzamiento de la versión 1.0.
- Modelos de FMU que se pueden emplear: Solo acepta FMUs de tipo co-simulation (CS).
- Generación: Hay que trabajar con la interfaz de usuario e ir importando cada FMU.

#### 4.6.2.2. Open Simulation Platform (OSP)

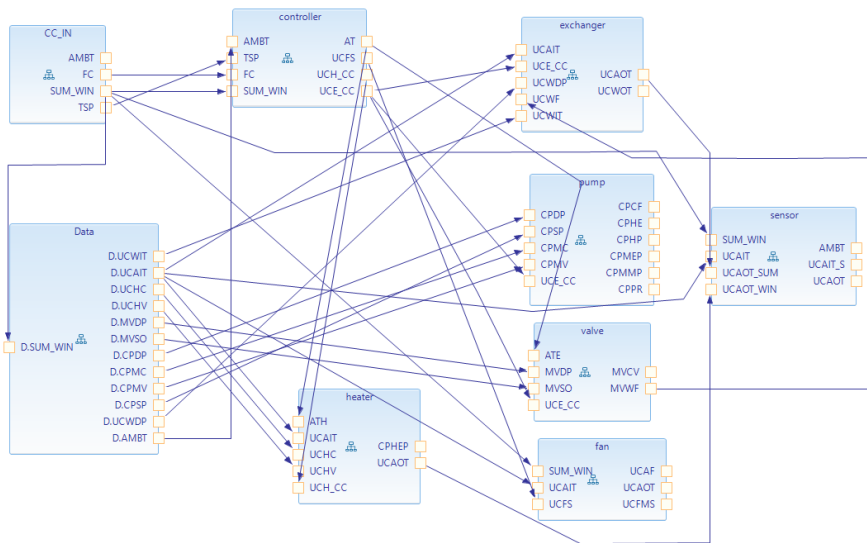
La plataforma Open Simulation Platform (OSP) (OSP, 2018) ha sido el resultado de la colaboración de Rolls-Royce Marine, NTNU (Norwegian University of Technology Science), DNV GL, y Hyundai Heavy Industries. Estas entidades participan en un proyecto industrial conjunto (OSP-JIP) para permitir las simulaciones de gemelos digitales y resolver los retos de diseño, en cuanto a operación y aseguramiento de sistemas complejos e integrados.

Dentro de este proyecto se ha desarrollado un conjunto de librerías para la co-simulación de FMU (Ghimire et al., 2021; Perabo et al., 2020). La plataforma OSP tiene dos formas de integrar FMUs para co-simulación, una basada en el estándar SSP, trabajando solamente con el fichero SSD, y otra basada en el entorno gráfico Kopl, que genera un fichero xml con toda la información de la jerarquía de la co-simulación.

Se realizaron pruebas de co-simulación empleando el entorno gráfico: el procedimiento es similar al que se realizó en las herramientas basadas en SSP, en primer lugar, se cargan los FMUs, una vez cargados se conectan las

entradas y las salidas de cada uno de ellos. El proceso que se debe de seguir para integrar FMUs es el siguiente:

- Lo primero que se ha de hacer es crear una nueva tarea de co-simulación y crear una configuración dentro de esta tarea.
- Antes de importar los modelos FMU es necesario crear los bloques a los que se asignarán después los FMU.
- El siguiente paso será importar los FMUs y editar sus propiedades, es decir, las entradas, salidas, parámetros y unidades.
- Conectar las salidas de los FMUs con las entradas de las otras, de acuerdo con el modelo de simulación que se vaya a co-simular.
- Establecer el tiempo inicial y final de la simulación y paso de tiempo.
- Se ejecuta la co-simulación.
- Visualización y exportación de resultados de la co-simulación.



**Imagen 4-54: Esquema de la conexión de FMUs para simulación en el entorno kopl**

El modelo integrado se puede ejecutar en este entorno y visualizar gráficamente los resultados obtenidos. Los resultados para cada bloque FMU se almacenan en un fichero csv y también se genera un fichero XML con la información correspondiente a cada uno de los FMU. Se muestran a continuación,

a modo de ejemplo, las gráficas obtenidas en una simulación realizada (imágenes 4-55 y 4-56):

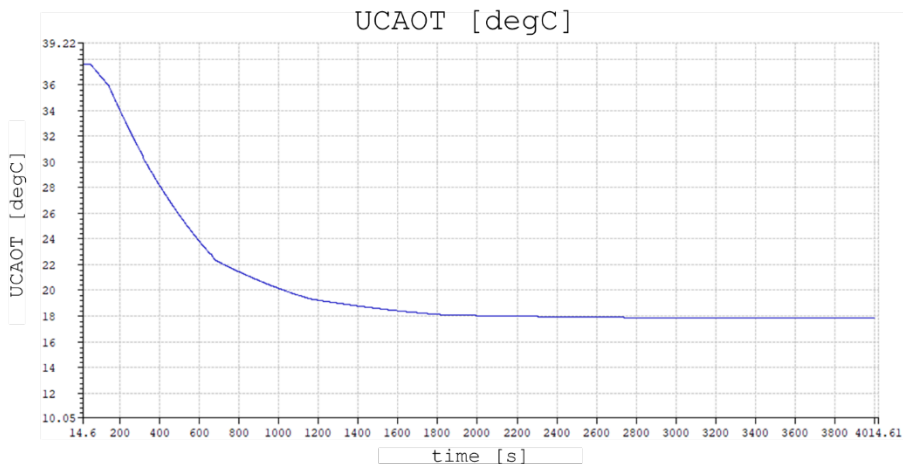


Imagen 4-55: Visualización de los resultados obtenidos para la variable UCAOT en el entorno Kopl

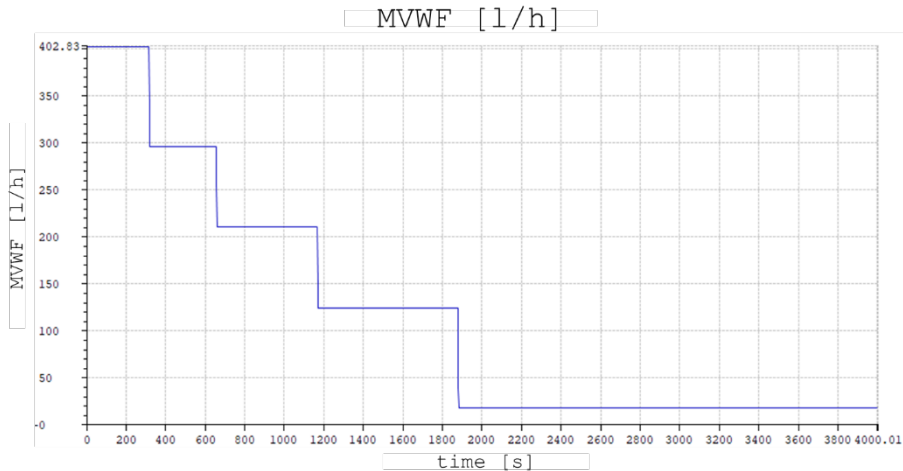


Imagen 4-56: Visualización de los resultados obtenidos para la variable MVWF en el entorno Kopl

Además del desarrollo a través de la interfaz gráfica, la herramienta Kopl de OSP también permite importar ficheros ssd con la jerarquía de componentes (FMUs) y conectores que los unen para la co-simulación. Se siguen los



pasos del estándar SSP, pero se trabaja únicamente con el archivo SSD, de modo que no se importa el SSP en su totalidad, sino solo la jerarquía.

Los aspectos analizados para evaluar la herramienta desarrollada por OSP, son los mismos que los que se valoraron en el caso de la herramienta SSP:

- Jerarquía: La jerarquía se establece en el entorno gráfico de Kopl y se describe en el archivo `OspSystemStructure.xml`. Si se emplea la herramienta SSD integrada en Kopl, la jerarquía se establece en el archivo `SystemStructure.ssd` (el mismo que usa el estándar SSP).
- Actualización: Si se emplea el entorno gráfico es necesario volver a importar el FMU que se haya modificado. Sin embargo, si se hace mediante código para cambiar un FMU es suficiente con aplicar las modificaciones en el código del fichero `SystemStructure.ssd`.
- Gestión de conectores: Se lleva a cabo mediante conectores directos (en la herramienta gráfica) o mediante código, también en el fichero `SystemStructure.ssd`.
- Gestión de parámetros: Parametrización de variables internas en FMUs individuales en el entorno “Kopl”.
- Gestión de unidades: Las unidades se definen y/o editan en el entorno Kopl para cada FMU.
- Herramientas de integración: La plataforma OSP es en sí misma una herramienta de integración y simulación de FMUs
- Entornos de simulación: La plataforma OSP es un entorno de simulación de FMUs.
- Actualizaciones: La OSP soporta el estándar FMI 1.0 y 2.0. Y aseguran que soportará el FMI 3.0 cuando esté operativo (actualmente en versión beta).
- Modelo de FMU: Puede aceptar tanto modelos de co-simulación como model exchange.
- Generación: En la interfaz gráfica es necesario trabajar con la interfaz de usuario “Kopl” e ir importando cada FMU. Si se trabaja con SSD se puede automatizar mediante programación en Python.

#### 4.6.2.3. FMUContainer. FMU integrador

El integrador de FMUs es, en resumen, un FMU formado por los diferentes FMUs que requieren ser interconectados entre sí para formar un sistema de simulación. Su objetivo es posibilitar al usuario final la gestión y la integración de modelos FMUs para la simulación de un sistema complejo formado por subsistemas. El resultado de esta integración es un nuevo FMU estándar que cumple la validación de FMI.

Se ha analizado la librería FMUContainer (CATIA-Systems, 2017) de Python perteneciente a la librería FMPy . FMUContainer permite, mediante la programación de un script, unificar todos los FMU que forman parte de un sistema que se quiera integrar. Esta librería facilita al usuario el control sobre las características de los FMUs a unir, conectores (entradas y salidas de las FMUs), parámetros (variables internas, constantes que no se pueden poner como entradas, etc.), unidades, y la dependencia entre los conectores. Como resultado de esta integración de FMUs se obtiene un FMU que cumple con todos los estándares del FMI y que por lo tanto se puede ejecutar en todos los entornos que trabajen con el estándar FMI.

Aspectos importantes para destacar sobre el integrador de FMUs:

- Jerarquía: La jerarquía de los FMUs interconectados se define de acuerdo con los requerimientos del usuario basados en el estándar FMI. Se describe en el archivo XML (modelDescription.xml).
- Actualización: Si se actualiza un FMU, solo es necesario conectar las entradas y/o salidas del nuevo FMU con los FMUs ya existentes.
- Gestión de conectores: La gestión de conectores, unidades y la dependencia entre esos conectores para su correcta trazabilidad se gestiona en el XML del FMU resultante.
- Gestión de parámetros: Parametrización de variables internas en FMU individuales y en el FMU integrador.
- Gestión de unidades: Las unidades vienen definidas en el XML "modelDescription.xml" de los modelos FMUs a unir, el FMU integrador las define en funciones de las unidades básicas en su archivo XML resultante.

- Herramientas de integración: Existe una amplia gama de softwares que son capaces de generar FMUs.
- Entornos de simulación: Se ejecuta en cualquier entorno que trabaje con FMUs.
- Actualizaciones: El FMI se actualiza con frecuencia, siendo la versión 2.0.3 la disponible y la versión 3.0 beta en prueba.
- Modelo de FMU: Integra FMUs tanto de tipo Model Exchange (ME) como las de tipo Co-simulation (CS).
- Generación: Se puede automatizar mediante programación en Python, usando la librería FMUContainer de FMPy.

#### 4.6.2.4. Conclusiones sobre la integración de FMU

Los desarrollos orientados a la integración y orquestación de FMUs son una importante herramienta para la co-simulación de FMUs, en especial, en el caso de sistemas complejos, donde establecer una jerarquía de simulación, homogeneizar variables y unidades, y posibilitar la actualización de modelos a lo largo del tiempo, pueden suponer importantes ahorros de trabajo si se compara con la simulación tradicional de sistemas complejos, o co-simulación de modelos sin considerar una metodología integradora.

En general, las herramientas de simulación con las que se generan los FMU permiten exportar el modelo con las unidades correspondientes (excepto en el caso de los FMUs generados en Amesim con entradas y salidas de tipo señal, no podrán llevar unidades definidas, solo los llevarán las variables internas que se hayan expuesto en el momento de compilar el FMU).

Los FMUs tienen dependencia de los binarios para su ejecución, esto es, para poder simular un modelo FMU en un sistema operativo dado, los binarios del FMU deben de ser compatibles con dicho sistema operativo. Como consecuencia, hay que tener en cuenta que los FMUs que se van a agregar deben de tener, como mínimo, un sistema operativo y una arquitectura en común. Dicha intersección de sistemas operativos y arquitectura determinará los sistemas operativos y arquitecturas en los que se podrá ejecutar el modelo FMU resultante.

Se pueden resumir los resultados obtenidos de la comparativa entre FMU-Container, SSP y OSP en la siguiente tabla:

**Tabla 4-4-5: Comparativa de funcionalidades en las herramientas analizadas para co-simulación de modelos FMU**

	FMUContainer	SSP	OSP
Jerarquía	✓	✓	✓
Cambio de FMU	✓	✓	✓
Gestión de conectores	✓	✓	✓
Gestión de parámetros	✓	✗	✓
Gestión de unidades	✓	✓	✓
Herramientas de integración	✓	✗	✓
Entornos de simulación	✓	✗	✓
Actualizaciones	✓	✗	✓
Modelo de FMU (Me y CS)	✓	✗	✓
Generación	✓	✗	✓

De la tabla anterior se deduce que las funcionalidades que aportan el FMUContainer y el SSP son similares, mientras que el SSP se ve penalizado en varios aspectos, entre los que destacan la falta de actualizaciones en el estándar, el hecho de que, al menos en el caso del OMSimulator, solo se pueden ejecutar modelos creados en la misma plataforma o, que no permita emplear FMUs de tipo model Exchange.

## 5. APLICACIÓN IA AL GEMELO DIGITAL

---

### 5.1. INTRODUCCIÓN

El caso de uso que se presenta en este capítulo tiene como objetivo analizar la implementación de algoritmos de aprendizaje dentro del entorno del gemelo digital. Se ha seleccionado como caso concreto de aplicación un sistema basado en inteligencia artificial, IA, para la detección de fallos en un sistema del buque, en este caso el de lucha contra incendios.

Aunque se describirá en detalle más adelante, la idea es emplear las señales de los sensores de presión del sistema de contraincendios (CI) por agua salada del buque, y analizarlos empleando técnicas de IA, que permitan identificar una situación de funcionamiento normal del sistema y cuando está sucediendo una rotura. Además, se planteará la metodología a seguir para, además de detectar el fallo, tratar de acotar su localización.

### 5.2. DETECCIÓN DE FALLOS EN TUBERÍAS

La detección de fallos en sistemas de tuberías es un problema que se ha planteado en muchos campos de la ingeniería. Lograr una detección rápida y determinar la ubicación de una posible rotura o estrangulamiento en los sistemas de tuberías ha sido objeto de numerosos desarrollos. Existen diferentes métodos para la detección de fallos en sistemas de tuberías, basados en lecturas de los sensores que componen el sistema: análisis de vibraciones, refractómetros acústicos, métodos basados en impedancia, tecnología radio-isótopo, etc.

Las técnicas tradicionales para la detección de problemas en tuberías, utilizadas en los sistemas de distribución de agua, pero también en el caso de los hidrocarburos, se clasifican normalmente en 3 grandes grupos: aquellos métodos basados en detección biológica, los basados en software y los basados en hardware (J. Zhang, 1997). Los métodos de detección biológica son aquellos que se basan en la mera inspección de la red de tuberías. Los procedimientos basados en software emplean diferentes técnicas computacionales, mientras

que los métodos basados en hardware se clasifican de acuerdo con los sensores y equipos utilizados para la detección de fallos. En (Datta & Sarkar, 2016) se realiza un análisis de los principales métodos empleado en la detección de fallos y estrangulamientos de tuberías. Se pueden encontrar clasificaciones similares en diferente bibliografía relacionada especialmente con el sector del oil&gas (Lu et al., 2020).

La mayoría de los métodos basados en hardware se centran en el uso de sensores y sistemas, que en algunos casos implican un costo adicional de la instalación, por lo que la prioridad en este caso es detectar el fallo con los medios disponibles, por lo tanto, se limitará al uso de mediciones de presión en el sistema. Además, centrándose en la aplicación al sistema contra incendios del barco, se puede observar que, además, muchos de ellos no se recomiendan para su uso en entornos marinos, debido a los problemas generados en los sistemas que funcionan con agua de mar.

El uso de métodos basados en software presenta la importante ventaja de que requerirá una mínima o nula inversión en equipamiento adicional (ya se trate de sensores adicionales o de otro tipo de equipamiento). Aunque el uso de métodos basados en software también presenta alguna desventaja, relacionada casi siempre con el tiempo necesario para su desarrollo y puesta en funcionamiento, ya sea en tiempo de desarrollo de modelos de simulación que sean capaces de replicar con suficiente fiabilidad el sistema o tiempo para toma de datos en el caso de modelos que se basen en datos reales de funcionamiento (por ejemplo, algoritmos de machine learning).

Del análisis realizado acerca de las diferentes alternativas para la detección de fallos en tuberías, se pueden hacer los siguientes comentarios:

- Se han encontrado diferentes desarrollos que se basan en registros y mediciones de presión en los sistemas de tuberías, los cuales emplean diferentes métodos, estando en su gran mayoría orientados a la detección de fallos en tuberías de distribución de agua potable (Abdulshaheed et al., 2017; Berardi et al., 2015; Giustolisi et al., 2015; Sousa et al., 2015).
- Es importante señalar que llevar a cabo un planteamiento similar al realizado en el capítulo 4, es decir, considerar la realización de un modelo de simulación capaz de replicar el funcionamiento real del

sistema, podría considerarse de interés. De hecho, existen múltiples estudios que contemplan simulación para detección de fallos en tuberías (Bermúdez et al., 2018; Meseguer et al., 2015; Oyedeko & Balogun, 2015; Verde et al., 2014; Q. Zhang et al., 2019). Si bien, dado que la finalidad de este capítulo es analizar las vías disponibles para la implantación de sistemas del buque inteligentes, y cómo llevar a cabo la implementación de algoritmos de este tipo en el buque, además de identificar las carencias que pueda haber actualmente para poder avanzar en esta línea, resulta razonable considerar el uso de técnicas basadas en IA/machine learning/redes neuronales que permitan dotar al sistema (o al gemelo digital del sistema) de inteligencia.

- Existen ejemplos de uso de IA en detección de fallos de tuberías, Laurentys presenta el desarrollo de un sistema experto para detección de fallos (Laurentys et al., 2011). Pero, sin embargo, todo parece apuntar que la tendencia es aprovecharse de las ventajas que las redes neuronales pueden presentar para la resolución de problemas de este tipo (analizar series de datos, reconocimiento de patrones). Como ejemplo de lo que se afirma, tanto Sukarno como Zhao emplean redes neuronales para la detección de problemas en sistemas de tuberías submarinas (Sukarno et al., 2007; Zhao et al., 2014).
- No se ha encontrado ninguna referencia en la que se apliquen RN a los sistemas de buques.

### **5.2.1. Detección de fallos en tuberías basada en inteligencia artificial**

Tal y como se ha comentado en el apartado anterior, en el campo del oil&gas y en el caso de sistemas de distribución hidráulica, ya existen estudios relacionados con la aplicación de redes neuronales para la detección de fallos. Se presenta a continuación una tabla-resumen realizada a partir de la información presentada por Pérez-Pérez donde se recopilan estudios recientes relacionados con este campo (Pérez-Pérez et al., 2021).

**Tabla 5-1: Referencias recientes uso de redes neuronales para detección de fallos en tuberías. Basado en (Pérez-Pérez et al., 2021)**

<b>Referencia</b>	<b>Información relevante</b>
(Leu & Bui, 2016)	Basada en: datos de presión Aplicado a sistema de distribución de agua Red neuronal: Aprendizaje Bayesiano
(Zadkarami et al., 2016)	Basada en: datos de presión y de caudal Aplicado a sistema de distribución de crudo (simulado) Red neuronal: Perceptrón multi-capas
(Zadkarami et al., 2017)	Basada en: datos de presión y de caudal Aplicado a sistema de distribución de crudo (simulado) Red neuronal: Clasificador con Perceptrón multi-capas Dempster-Shafe
(Gómez-Camperos et al., 2019)	Basada en: datos de caudal Aplicado a sistema de distribución de agua (experimental) Red neuronal: Perceptrón multi-capas
(Jia et al., 2018)	Basada en: datos de presión Aplicado a sistema de distribución de agua (experimental) Red neuronal:
(Pulido et al., 2019)	Basada en: datos de presión Aplicado a sistema planta de vapor (experimental) Red neuronal: Algoritmo backpropagation
(Kang et al., 2018)	Basada en: datos de presión Aplicado a sistema de distribución de agua (experimental) Red neuronal: Red Neuronal en el Espacio de Estados
(Zhao et al., 2014)	Basada en: datos de presión Aplicado a sistema de distribución de agua (simulado) Red neuronal: Red convolucional
(Javadiha et al., 2019)	Basada en: datos de presión Aplicado a sistema de distribución de agua (experimental) Red neuronal: Red convolucional y razonamiento Bayesiano



En el capítulo 2 se adelantaron algunos conceptos relacionados con la inteligencia artificial y con el machine learning. En este capítulo se pasará directamente a detallar en qué consisten las redes neuronales.

### 5.3. CONCEPTOS BÁSICOS SOBRE REDES NEURONALES

Las redes neuronales son un modelo computacional que trata de simular el funcionamiento de las neuronas del cerebro humano, con el objetivo de resolver problemas computacionales complejos que no son fácilmente programables. En estos casos lo que se hace es “enseñar” o, para ser más exactos, “ayudar a aprender” a esta red neuronal artificial.

#### 5.3.1. La Neurona. Perceptrón

En una red neuronal, la neurona representa la unidad mínima de procesamiento. Cualquier red neuronal estará formada por estos componentes elementales de proceso. En cada instante, una neurona estará caracterizada por un determinado valor (o estado de activación). Además, existirá una función de activación asociada a cada neurona que se encarga de transformar el estado de activación en una señal de salida. El esquema general de una neurona (con 3 señales de entrada) es el siguiente:

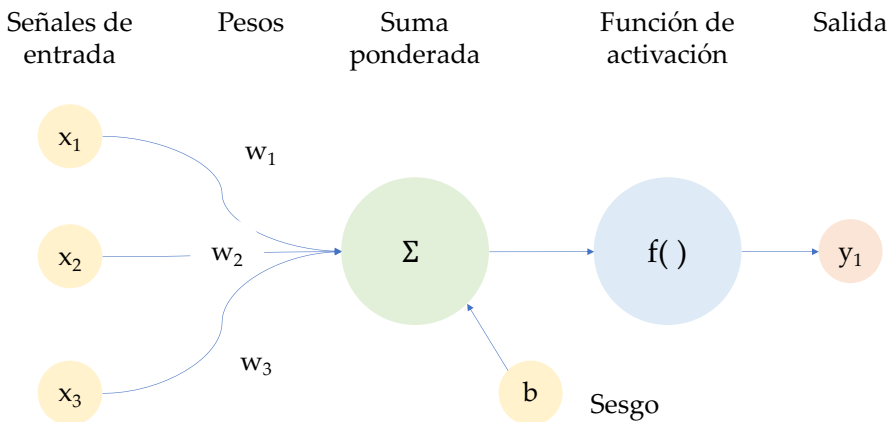


Imagen 5-1: Esquema general de una neurona

Donde  $x_1$ ,  $x_2$  y  $x_3$  son los valores de entrada y  $w_1$ ,  $w_2$  y  $w_3$  son los pesos, que definen con qué intensidad afecta la variable de entrada a la neurona. Dentro de la neurona se lleva a cabo una suma ponderada de los valores de entrada:

$$h_i = \sum_i x_i w_i + b \quad (5-17)$$

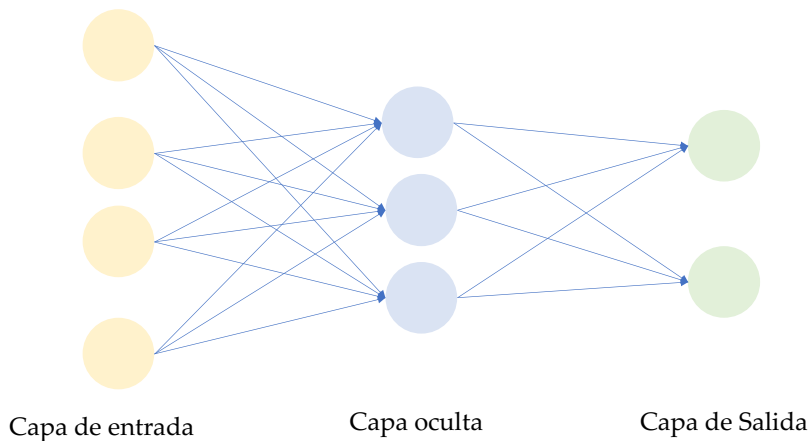
y la función de activación representa simultáneamente la salida de la neurona y su estado.  $y_1$  es el valor de salida de la neurona. Una neurona procesa un vector de entradas y produce una respuesta o salida única.

$$y_i = f(\sum_i x_i w_i + b) \quad (5-18)$$

Como se puede observar, la operación interna que realiza la neurona es similar a un modelo de regresión lineal, donde  $b$  sería el término independiente. En este caso es el sesgo (bias).

### 5.3.2. Red neuronal

Se denomina red neuronal a aquella estructura constituida por varias neuronas asociadas. En general, la forma de asociarse será en forma de capas, de forma que una red neuronal estará formada por varias capas (al menos una capa de entrada, una capa oculta y una capa de salida) y estas capas formadas por una o varias neuronas.



**Imagen 5-2: Esquema general de una red neuronal**

La capa de entrada está compuesta por las neuronas que reciben las señales de entrada mientras que la capa de salida será aquella cuyas neuronas dan la respuesta como salida. La capa (o capas) oculta(s) no tienen relación directa con el entorno.

El conocimiento en una red neuronal está jerarquizado, es decir, cada capa puede especializarse en una cosa y, además, a medida que se va avanzando en las diferentes capas el conocimiento que va adquiriendo la red puede ser más complejo. Si se van añadiendo capas, el conocimiento va adquiriendo complejidad. Si la red está formada por más de una capa oculta, lo habitual es referirse a Deep learning o aprendizaje profundo.

### 5.3.2.1. Funciones de activación

Las funciones de activación de redes neuronales artificiales son funciones no lineales, necesarias en redes neuronales multicapa, ya que, si no existieran funciones de activación a la salida de cada neurona, todas las capas de la red se podrían reducir a una capa, puesto que la red estaría compuesta por una suma de regresiones lineales (lo que equivaldría a una única regresión lineal).

Una función de activación introduce una no-linealidad, de forma que, al concatenar neuronas dentro de una red, no se puede reducir a una única neurona. Existen diferentes funciones de activación, cada una de ellas recomendada para un propósito concreto, por ejemplo, si se quiere predecir la probabilidad de un resultado o en el caso de una regresión logística donde la clasificación es 0 o 1, la función sigmoide (la cual se describirá a continuación) sería la más adecuada.

Las principales funciones de activación son las siguientes:

- Sigmoid – Sigmoide: La función sigmoide transforma los valores a un rango (0,1), donde los valores altos tenderán de manera asintótica a 1 y los valores muy bajos lo harán a 0.

$$f(x) = \frac{1}{1 + e^{-x}} \quad (5-19)$$

- Tanh – Tangente hiperbólica: Esta función transforma los valores introducidos a una escala (-1,1), donde los valores altos tenderán de forma asintótica a 1 y los valores muy bajos a -1.

$$f(x) = \frac{2}{1 + e^{-2x}} - 1 \quad (5-20)$$

- ReLU – Rectified Lineal Unit. Lo que hace la función ReLU es transformar los valores de la siguiente forma: los valores negativos los anula mientras que los positivos los deja igual que como han entrado

$$f(x) = \max(0, x) = \begin{cases} 0 & \text{para } x < 0 \\ x & \text{para } x \geq 0 \end{cases} \quad (5-21)$$

- Leaky ReLU – Rectified Lineal Unit: Esta función toma los valores de entrada y deja los positivos como están, mientras que los negativos los transforma multiplicándolos por un coeficiente rectificativo.

$$f(x) = \max(0, x) = \begin{cases} a \cdot x & \text{para } x < 0 \\ x & \text{para } x \geq 0 \end{cases} \quad (5-22)$$

- Softmax – Rectified Lineal Unit: Esta función transforma las salidas a una representación basada en probabilidades, de tal forma que la suma de todas las probabilidades de la salida es igual a 1.

$$f(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad (5-23)$$

### 5.3.3. Tipología de red seleccionada

Dado que no es objeto de este trabajo definir en detalle lo que son las redes neuronales y las diferentes tipologías, entrenamientos o algoritmos de aprendizaje que estas pueden emplear, este apartado se limitará a tratar las características concretas de la red empleada en este caso de uso.

Se ha seleccionado una red neuronal a la que se conoce como feed-forward fully connected. Es la tipología de red más ampliamente empleada, y sobre la que se aplican los más conocidos algoritmos de aprendizaje como es el método de backpropagation. Se dice que una red es feed-forward, cuando las neuronas de las diferentes capas no se conectan entre sí, pero si se conectan entre una capa y la siguiente. Además, cuando todas las neuronas de una capa están conectadas con todas las neuronas de la capa anterior, y de la siguiente, se dice que esta es “fully connected”.

### 5.3.4. Entrenamiento de redes neuronales

Como se comentó en el capítulo 2, existen 3 tipos de aprendizaje para los algoritmos de machine learning: aprendizaje supervisado, aprendizaje no supervisado y aprendizaje reforzado.

Por las características de los datos de los que se va a disponer en el caso de uso considerado parece razonable emplear un aprendizaje supervisado: aquel en el que al algoritmo se le suministran os los datos previamente etiquetados,

es decir, las entradas y las salidas correspondientes. A partir del conocimiento de estos problemas resueltos, el algoritmo será capaz de resolver nuevos problemas. Un ejemplo de aprendizaje supervisado serían las redes neuronales de tipo backpropagation.

#### 5.3.4.1. Algoritmo de Backpropagation y Descenso del Gradiente

El algoritmo de backpropagation se emplea junto al descenso del gradiente. El método de backpropagation permite, como se describirá más adelante, calcular las derivadas parciales de cada uno de los parámetros de la red con respecto a la función de coste (la función de coste es aquella función que permitirá evaluar el error de la red). Esas derivadas parciales serán uno de los componentes que se necesitarán para poder optimizar una red neuronal usando el descenso del gradiente.

El descenso del gradiente, cuyo funcionamiento también se explicará a continuación, permite optimizar la red minimizando el coste de esta, es decir, minimizando el vector gradiente (el cual estará formado por las derivadas parciales de los parámetros de la red respecto al coste, calculadas con el algoritmo de backpropagation).

#### 5.3.4.2. Algoritmo de Backpropagation

El Backpropagation es un algoritmo de aprendizaje, el cual nace en el año 1986, después del invierno de la inteligencia artificial (Rumelhart et al., 1986). Mediante el uso del algoritmo de backpropagation se consigue que una red neuronal sea capaz de autoajustar sus parámetros. El error se propaga hacia atrás en toda la red, responsabilizando a cada neurona del porcentaje de error que le corresponde. De este modo se puede saber cuánto hay que modificar cada parámetro de dicha neurona.

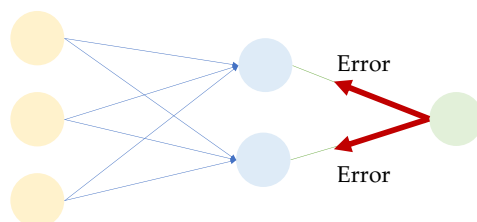


Imagen 5-3: Propagación del error en el algoritmo de backpropagation

### 5.3.4.3. Método de descenso del gradiente

El método del descenso del gradiente es común a gran parte de los algoritmos de machine learning, y se emplea para conseguir minimizar la función de coste de la red y por lo tanto el error, es decir, conseguir entrenar dicha red.

Considerando una función tridimensional (asumiendo que esta función podría representar el error en función de 2 parámetros, aunque, en el caso de una red neuronal se podría tener miles de parámetros), se conocerá como gradiente al vector resultante, formado por las derivadas parciales de la función respecto a cada uno de los parámetros. Si se consigue minimizar ese vector gradiente, eso significará que se ha localizado un mínimo de la función error. Para ello es necesario seguir los siguientes pasos:

1. Se parte de un punto aleatorio (inicialización).
2. A continuación, es necesario localizar la mayor pendiente en la posición actual, que será la mayor pendiente en cada una de las dos direcciones. El vector resultante de esos 2 vectores determinará el gradiente. Como el objetivo es minimizarlo, lo que se hará será desplazarse en esa dirección en el sentido negativo (descendiendo).
3. Es necesario seguir haciendo el mismo proceso hasta que casi no exista variación entre una iteración y la siguiente (pendiente nula). Significará que se ha convergido, por lo tanto, se ha alcanzado un mínimo local.
4. El último paso del proceso consiste en añadir a lo anterior la ratio de aprendizaje, el cual indica cuánto afecta el gradiente a la actualización de los parámetros en cada iteración (o lo que es lo mismo, cuánto se avanza en cada paso que se va dando).

### 5.3.4.4. Algoritmos de Levenberg-Marquardt, regularización bayesiana y escalado conjugado del gradiente para backpropagation

El algoritmo de entrenamiento Levenberg-Marquardt es una función de entrenamiento para redes neuronales que actualiza los valores de peso y sesgo según la optimización de Levenberg-Marquardt, también conocido como el método de mínimos cuadrados amortiguados (DLS) (Gavin, 2019; Sapna, 2012). Se trata del algoritmo más comúnmente empleado en backpropagation,

debido a su rapidez de convergencia con errores suficientemente bajos. Sin embargo, alcanzar errores pequeños no garantiza la capacidad de generalización del modelo, produciendo en este un problema de sobreentrenamiento, lo cual significa que la red presenta un error muy bajo con los patrones de entrenamiento, pero el error aumenta con los patrones de validación.

Las redes neuronales entrenadas empleando Levenberg-Marquardt presentan el inconveniente que, sobre todo en el caso de que la capa oculta tenga un elevado número de neuronas, se puede presentar sobreentrenamiento de la red. Este fenómeno consiste en que la red neuronal es capaz de aprender el ruido que se pueda presentar en la señal, que como puede suceder en los sensores de los sistemas de un buque, es un fenómeno habitual. El sobreentrenamiento es un fenómeno indeseado, ya que puede perturbar los resultados de la red cuanto se esté trabajando con datos que difieren de los casos de entrenamiento. En general, suele surgir por dos motivos: el uso de una red neuronal demasiado compleja, o un valor excesivo para los pesos,  $w$ .

Una forma de evitar el sobre-entrenamiento que a veces se produce en Levenberg-Marquardt consiste en sustituir este método por la regularización bayesiana (Foresee & Hagan, 1930; Lopez & Caicedo, 2005; MacKay, 1992). En realidad, la regularización bayesiana es una función de entrenamiento que actualiza los pesos y sesgos de acuerdo con la optimización de Levenberg-Marquardt, pero con la característica de que consigue minimizar una combinación de errores cuadrados y pesos y determina la correcta combinación para producir una red que generalice bien.

Finalmente, se conoce como scaled conjugate gradient para backpropagation a aquella función de entrenamiento que actualiza los pesos y los sesgos por el método del escalado conjugado del gradiente. Se desarrolló con el objetivo de acelerar la convergencia respecto a los métodos anteriores.

## **5.4. DESCRIPCIÓN DEL SISTEMA EMPLEADO COMO CASO DE USO**

El sistema de contraincendios con agua salada suministra agua salada a presión a los boquiles contraincendios del buque. Realizará además el suministro de agua salada a presión para otros servicios especializados de lucha contraincendios (estación de espuma o agua nebulizada), así como a otros

sistemas y consumidores de agua salada en el buque. Se trata de un sistema presurizado que será alimentado con agua salada por medio de cuatro bombas centrífugas de 200 m<sup>3</sup>/h y 12 bar, diseñadas para funcionamiento continuo, dotadas con regulador de velocidad y que aspirarán directamente desde los respectivos colectores de agua salada situados en la misma cámara en la que se encuentren localizadas.

El buque empleado como base para este caso de uso está dividido transversalmente por un mamparo en la cuaderna 66, que divide el buque en dos zonas principales de fuego. Las bombas contraincendios se encuentran situadas bajo la línea de flotación, con el siguiente emplazamiento:

- Dos bombas en la Cámara de Auxiliares.
- Una bomba en la Cámara de Máquinas de Proa.
- Una bomba de la Cámara de Máquinas de Popa.

Cada zona de fuego cuenta con 2 bombas con una capacidad de: 200 m<sup>3</sup>/h y 12 bar (400 m<sup>3</sup>/h en cada zona). Además, las bombas de C. I. deben tener la capacidad suficiente para suministrar el caudal máximo requerido por el sistema con una bomba fuera de servicio. El caudal máximo en esta condición es de 600 m<sup>3</sup>/h. El anillo del sistema contraincendios estará constantemente presurizado, siendo suficiente el funcionamiento de una bomba para mantener la presión de 7 kg/cm<sup>2</sup> (6.87 bar) en el punto más alto del circuito.

El sistema se dispondrá formando un sistema de anillo vertical, el cual estará subdividido en dos colectores principales horizontales. El colector superior estará situado sobre la cubierta principal y el colector inferior bajo la segunda cubierta. Cada uno de los dos colectores horizontales longitudinales que conforman el anillo de C.I. tiene la capacidad suficiente que permita, en condición de emergencia suministrar el caudal total requerido por el sistema.



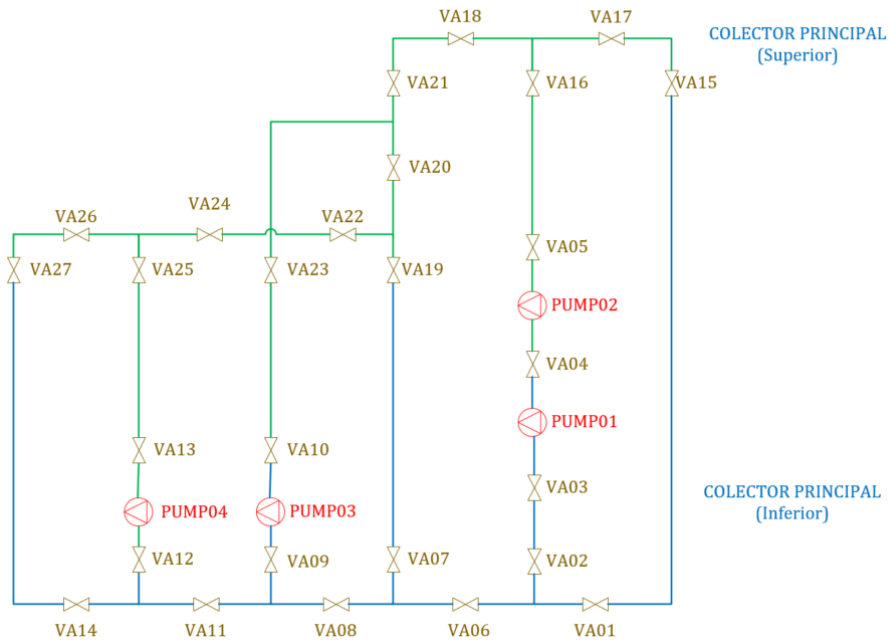


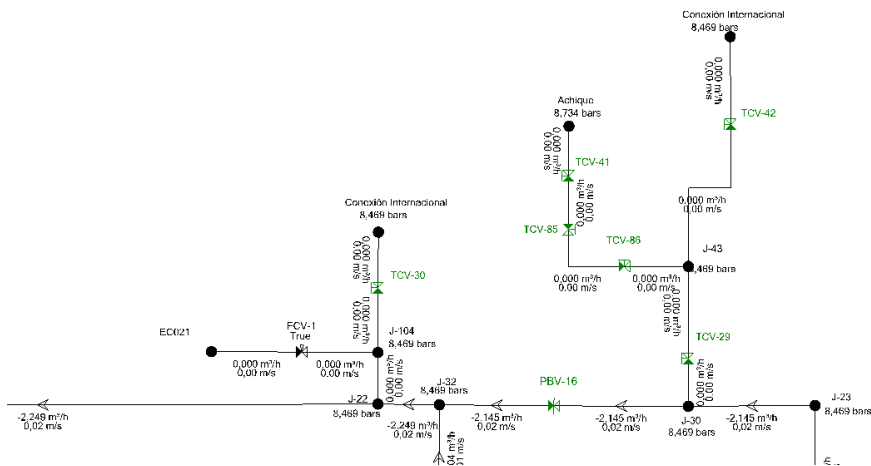
Imagen 5-4: Esquema simplificado del sistema contraincendios

## 5.5. MODELOS NUMÉRICOS. SIMULACIONES

### 5.5.1. Descripción general del modelo para el sistema contraincendios

Se desarrolló un modelo para simulación del comportamiento hidráulico del sistema contraincendios basado en el sistema real del buque base. Se optó por plantear un modelo de simulación de este sistema, en las primeras fases de desarrollo de este trabajo, para tratar de conocer y caracterizar el funcionamiento del sistema y poder disponer, en poco tiempo, de unos datos preliminares que pudiesen servir para al menos plantear la estructura de la red neuronal, de forma que al disponer de datos reales, los cambios a realizar en la red fueran mínimos (probablemente modificar el tamaño de la red o el algoritmo de entrenamiento) y realizar un primer entrenamiento con ella (aun considerando que los datos empleados serán basados en un modelo de simulación simplificado). Si bien, se ha de destacar que el desarrollo de un modelo de simulación en este caso de uso es poco relevante para el objetivo que se busca, y que es que es precisamente, validar el uso de modelos basados en datos frente al uso de modelos de simulación basados en física, como los empleados en el capítulo 4 (Erikstad & Ove, 2017).

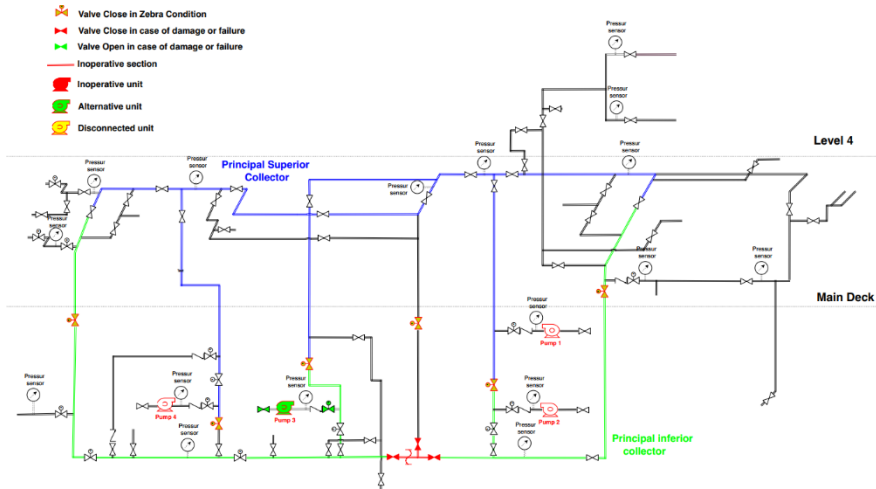
Para el desarrollo de este modelo del C.I. del buque se ha empleado un software comercial que permite simular circuitos hidráulicos en estados estacionarios, así como en regímenes transitorios. Se trata del software Watercad/Hammer de Bentley, un software diseñado especialmente para la simulación de sistemas de distribución de agua. Se replicó el sistema (tuberías, válvulas, bombas) de forma simplificada, empleando el diagrama funcional del sistema, así como los diferentes consumidores del sistema (monitores contra incendios, eyectores de achique, estaciones contra incendios, etc.) El software permite incorporar (aunque no se haga de manera gráfica) las características de las tuberías: materiales, dimensiones, e incluso accesorios (codos, té u otros elementos).



**Imagen 5-5: Ejemplo de ramales del modelo de simulación del sistema**

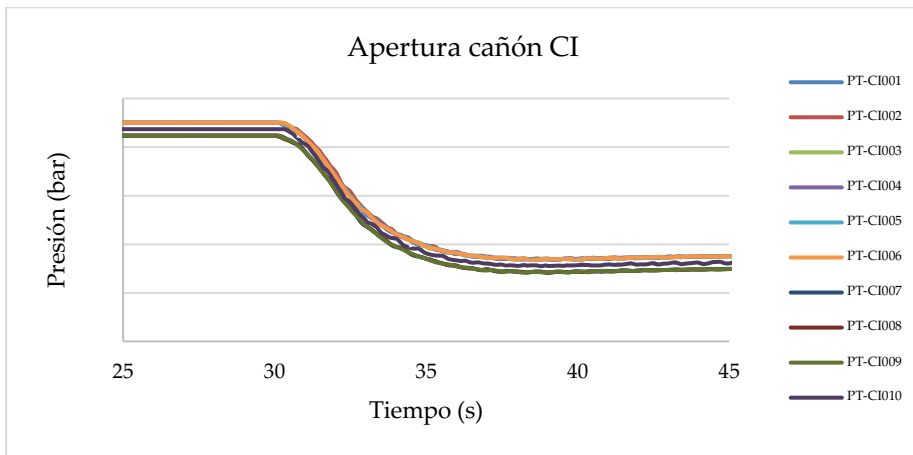
Con el modelo se llevaron a cabo la identificación de los modos de fallo del sistema (posibles tramos de tubería afectados y fallo en bombas), y se realizaron simulaciones de funcionamiento:

- Funcionamiento normal del sistema (apertura de consumidores de agua salada del sistema contra incendios).
- Posibles roturas de tuberías (por impacto) ubicados en diferentes tramos de los colectores principales o de ramales del sistema (según los modos de fallo identificados).

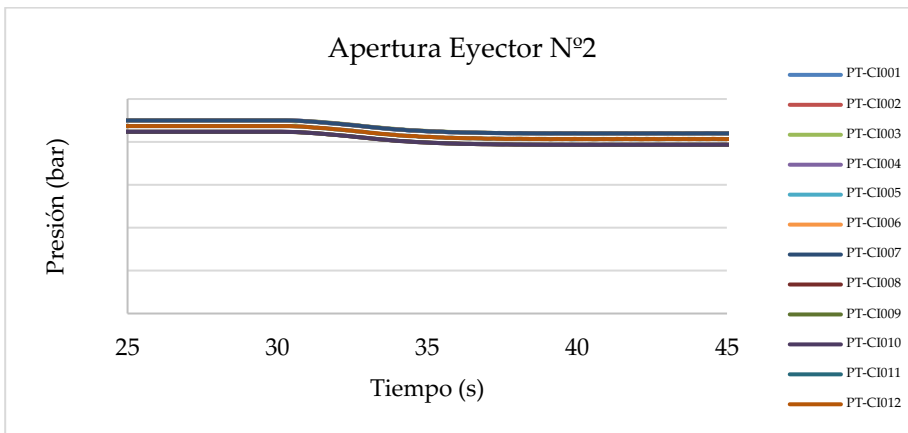


**Imagen 5-6: Modo de fallo. Identificación tubería dañada + sección inoperativa**

Los resultados obtenidos en estas simulaciones se reducen a los valores de presión en los nodos definidos en las posiciones en las que se encuentran los sensores de presión del buque, ya que se parte del punto en el que la única fuente de información que existirá en el buque real es la que ofrecen los sensores de presión existentes en el buque.



**Imagen 5-7: Resultados de la simulación (caída de presión) realizada para la apertura del cañón contraincendios**



**Imagen 5-8: Resultados de la simulación (caída de presión) realizada para la apertura del eyector de achique N°2**

## 5.6. PRUEBAS EN BUQUE REAL. FASE 1

Las primeras pruebas realizadas a bordo fueron en el buque empleado como buque base, un buque militar que se encontraba en la fase final de su construcción en el astillero de Navantia-Ferrol. Los objetivos principales de estas pruebas fueron los siguientes:

- En primer lugar, analizar de qué forma será necesario realizar la adquisición de datos en un buque real, de cara a pensar en la implementación de un sistema de este tipo en el entorno del gemelo digital del buque. Y no solo en lo referente al mecanismo para la adquisición de datos (el “cómo se van a recoger esos datos”) sino considerando todo aquello que sea necesario para que esos datos sean válidos, por ejemplo: qué frecuencia será necesaria en las señales o qué precisión habrá que alcanzar en los valores medidos.
- En segundo lugar, el principal objetivo fue obtener valores reales de funcionamiento del sistema.
- En tercer lugar, y dado que se dispone de un modelo de simulación simplificado, validar dicho modelo.

Las pruebas se dividieron en dos partes: Una primera en la que se trataron de caracterizar los principales consumidores de agua salada del sistema, como son: monitores C.I., eyectores de achique, estaciones C.I., rociadores y otros

elementos que toman agua de este mismo sistema (como puede ser el servicio de descontaminación, refrigeración de la bocina o planta de tratamiento de aguas residuales, elementos que tendrán una pequeña demanda de caudal). Para ello se llevaron a cabo repetidas aperturas/cierre de los mismos. Además, se aprovechó también para arrancar/parar las bombas del sistema, para tener datos de las curvas de presión que se generan en el arranque y en la parada de las bombas.

En la segunda parte de las pruebas se simuló la rotura de un tramo de tubería y el estrangulamiento parcial y total (colapso) de la misma. Circunstancias que pueden suceder si, por ejemplo, se produce un impacto en el que se vea afectado este sistema.

### **5.6.1. Preparación de las pruebas**

Se elaboró un procedimiento de realización de pruebas en el que se definió la realización de un mínimo de 4 pruebas:

- Prueba N°1 - Funcionamiento normal del sistema: En esta primera prueba se realiza el arranque y parada de bombas, además de abrir de forma secuencial consumidores del sistema, hasta que la caída de presión exige el arranque de una bomba adicional, con el objetivo de caracterizar las curvas de presión que se presentarán en el sistema cuando este funcione en condiciones normales.
- Prueba N°2 -Caracterización de consumidores. Apertura y cierre (independiente) de diferentes consumidores. Estos consumidores serán, en general: eyectores de achique, boquiles del sistema contraincendios, o dado que este buque cuenta con un monitor contraincendios, también se caracterizará la apertura de este monitor por ser el mayor demandante de agua salada de todo el sistema. Se realizará también la apertura/cierre de otros consumidores menores con una baja demanda de caudal, como puede ser el caso, por ejemplo, de la refrigeración de la bocina.
- Prueba N°3 -Simulación de rotura de tubería. Apertura “repentina” de una válvula instalada en el sistema para tal finalidad.
- Prueba N°4 -Simulación de estrangulamiento de tubería. Cierre parcial repentino de la válvula instalada para la realización de las pruebas N°3 y N°4.

Para simular la rotura (y estrangulamiento) de una tubería (pruebas N°3 y N°4) se llevó a cabo una instalación provisional: En la EC-CIS020, estación contraincendios de 1 ½, se retiró el boquil CI y se instaló una válvula de bola conectada a una manguera de bomberos con una lanza, tal y como se muestra en la imagen 5-9.



**Imagen 5-9: Instalación provisional para pruebas**

Con la lanza abierta, la apertura rápida de la válvula permitirá hacer una apertura “casi instantánea” (de cara a simular una rotura). Además, aperturas parciales de la válvula permitirán simular diferentes estrangulamientos causados por un posible impacto en la tubería. Se colocó una plantilla con los posibles grados de apertura de la válvula (de 0° a 90°) para poder detallar el ángulo de “estrangulamiento” empleado.

### **5.6.2. Realización de pruebas**

Parte de la realización de las pruebas se llevaron a cabo desde la sala de control desde donde se controló la apertura/cierre de consumidores. Los datos fueron registrados desde el sistema de control del buque a un dispositivo raspberry con una frecuencia de refresco del registro de 50 Hz. Para ello se programó un sencillo registrador de datos (datalogger) que permite recoger los datos del sistema de control del buque para su almacenamiento. De forma

análoga podría desarrollarse una toma de datos similar que en lugar de almacenar los datos para su posterior procesado (en este caso ese era el objetivo) permitiese el análisis de las series de datos en tiempo “útil”, es decir, resultaría relativamente sencillo de programar un mecanismo para la toma de datos, en este caso en la capa del sistema de control del buque, pero que podría ser desarrollada de forma similar en capas más bajas.

Se ha de destacar que la programación del sistema de registro y toma de datos fue desarrollada por personal de Navantia-Sistemas y se desconoce el protocolo empleado para tal finalidad. Aunque, de las pruebas realizadas en el caso de uso del capítulo 4, se conoce que el software empleado para el sistema de control de plataforma admite protocolos API Rest, que fueron los que se emplearon en el caso de uso del HVAC, donde se enviaba cada 2 segundos un string con la serie de datos necesaria, por lo que se deduce que en este caso se podría hacer algo similar.

Se llevaron a cabo un total de 7 pruebas: 4 de ellas orientadas a lo que podría ser el funcionamiento normal del sistema, y 3 en las que el objetivo era obtener mediciones en condiciones similares a lo que podría ser una rotura (y estrangulamiento) de tubería. Como se puede observar el número de pruebas propuestas inicialmente (en el procedimiento) era 4, pero finalmente fue posible repetir 3 de las pruebas. Se adjuntan a continuación, a modo de ejemplo, los resultados obtenidos en una de las pruebas realizadas para un funcionamiento normal del sistema y para simulación de una rotura de tubería.

- Sistema CI operando (prueba 1): En esta prueba el procedimiento fue el siguiente:
  - La prueba se inicia con la bomba 3 funcionando (9 bar).
  - Se abre uno de los cañones, los cuales tienen una demanda de 400 m<sup>3</sup>/h, por lo que se produce una caída drástica de presión.
  - Se cierra el cañón.
  - Se abren los eyectores de achique disponibles hasta que la presión cae (el caudal de los eyectores a 9 bar es de 45 m<sup>3</sup>/h).
  - Se cierran los eyectores en orden inverso a la apertura.

En la siguiente gráfica se pueden ver los valores de presión obtenidos en cada uno de los sensores del sistema en esta primera prueba:

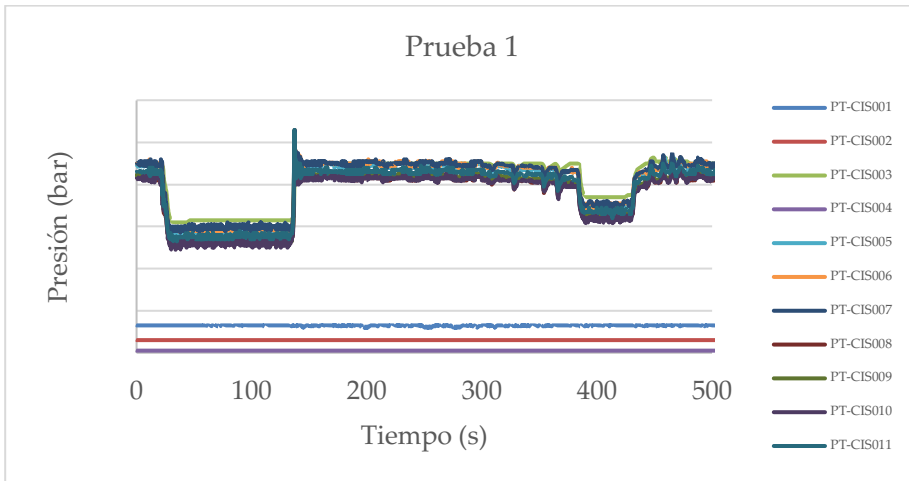


Imagen 5-10: Señales de presión recogidas en los sensores del sistema contraincendios por agua salada del buque (prueba 1)

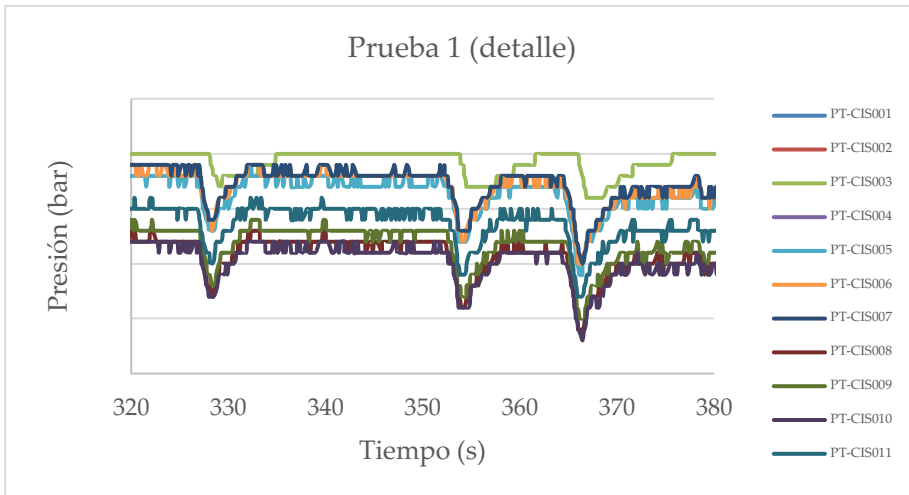
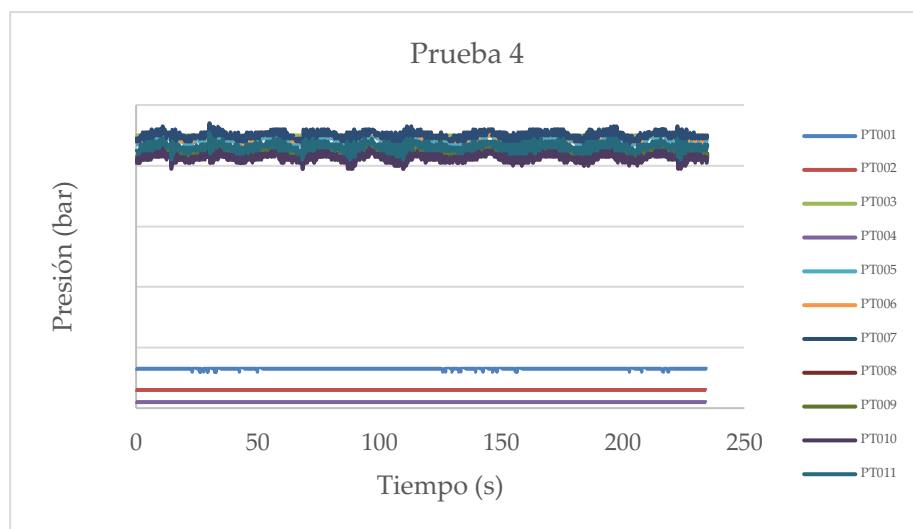


Imagen 5-11: Señales de presión recogidas en los sensores del sistema contraincendios por agua salada del buque (prueba 1). Vista de detalle

Como se puede apreciar, la bomba es capaz de recuperar después de cada apertura de los eyectores (de hecho, la apertura/cierre del primer eyector apenas se aprecia), y es con la apertura del último eyector (todos tienen la misma demanda de caudal), cuando la presión cae rápidamente por debajo de 8 bar. También se puede observar (en la imagen 5-11) que la resolución en la señal es muy limitada (décima de bar).



- Simulación de buque en problemas (prueba 4): Partiendo de la instalación abierta, se llevaron a cabo 4 aperturas/cierres (lo más rápido posible) de la válvula de apertura rápida colocada en la manguera. En la gráfica se pueden ver los resultados obtenidos en cada uno de los sensores:



**Imagen 5-12: Señales de presión recogidas en los sensores del sistema contraincendios por agua salada del buque (prueba 4)**

Como se puede apreciar en la gráfica existe una leve oscilación en la presión, pero dicha oscilación no encaja con la apertura/cierre del boquil de 1 ½. No fue posible obtener datos de caídas/incrementos de presión para las simulaciones de rotura/estrangulamiento de tuberías. Esto ha sido debido a: el bajo caudal del boquil que se abrió, el hecho de que las bombas del sistema sean de velocidad variable (de modo que tienden a recuperar rápidamente las caídas de presión), pero sobre todo por la baja resolución obtenida en la señal.

### 5.6.3. Conclusiones/resumen de las pruebas realizadas

Se llevaron a cabo pruebas en el sistema CI de un buque real. Realizando registros a 50 Hz en 11 sensores de presión del sistema. Las principales conclusiones del aprendizaje obtenido en la realización de las pruebas han sido las siguientes:

- La frecuencia de refresco de las señales ha sido de 50 Hz, valor suficiente para identificar patrones en las caídas de presión de los sensores. Sin embargo, la precisión de la medida ha sido de décima de bar, lo cual es insuficiente para obtener datos con una precisión que permita identificar correctamente patrones en las caídas de presión de los sensores.
- En cuanto a los resultados obtenidos, en el caso del funcionamiento normal del sistema, los resultados permiten identificar perfectamente los eventos (pero teniendo en cuenta que la precisión no es la necesaria para identificar los patrones de caída de presión). Sin embargo, en el caso de la "rotura" y "estrangulamiento" de la tubería, no ha sido posible identificar los eventos. Además de la falta de resolución obtenida, se considera que el haber empleado un boquil de 1 ½, unido al hecho de que las bombas del sistema CI son de velocidad variable ha hecho no se observe fluctuación en la presión del sistema cuando se abre/cierra dicho boquil.

#### 5.6.4. Validación del modelo de simulación

Aunque, como se acaba de detallar, los resultados obtenidos en las pruebas realizadas no permiten realizar el entrenamiento de la red neuronal, se aprovecharon estas pruebas para llevar a cabo la validación del modelo desarrollado. Para ello, se replicaron los eventos que se llevaron a cabo en las pruebas a bordo del buque. Se compararon los resultados obtenidos en las simulaciones en estado estacionario, como los ejemplos de las imágenes 5-13 y 5-14. En la Prueba 1 se llevó a cabo la apertura/cierre del monitor CI y posteriormente la apertura secuencial de los eyectores de achique (y el correspondiente cierre). En la prueba 3 se llevó a cabo la apertura/cierre de diferentes consumidores, aquellos que tienen poco caudal apenas muestran variación ninguna (o ninguna) en la presión del sistema, por lo que en las gráficas parece no verse nada hasta que se abre el monitor CI.

Es evidente que estas simulaciones no reflejan de manera realista el sistema, pero sirven para que, con pocas exigencias computacionales, se pueda validar el modelo desarrollado.

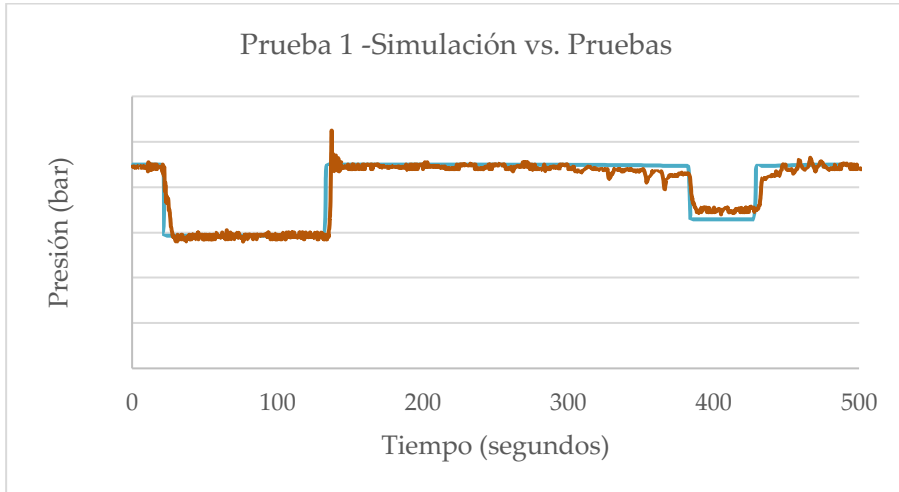


Imagen 5-13: Comparativa resultados (sensor PT007) real vs simulados (prueba1)

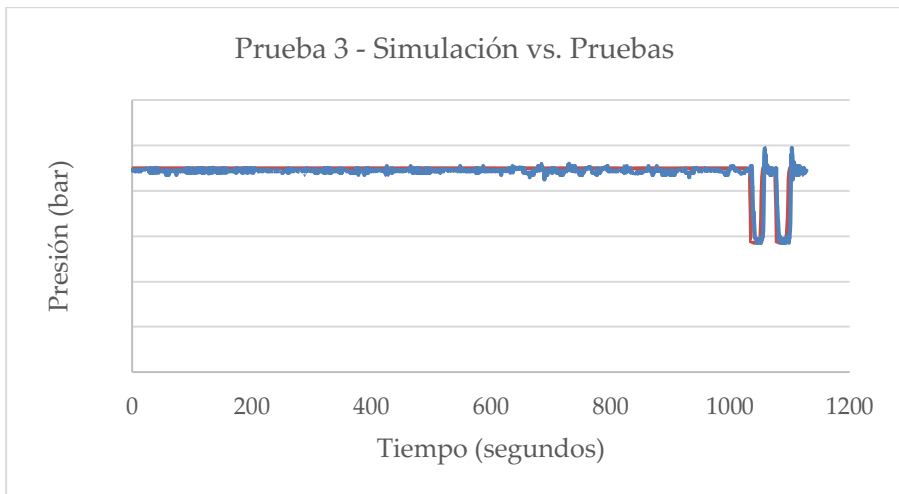
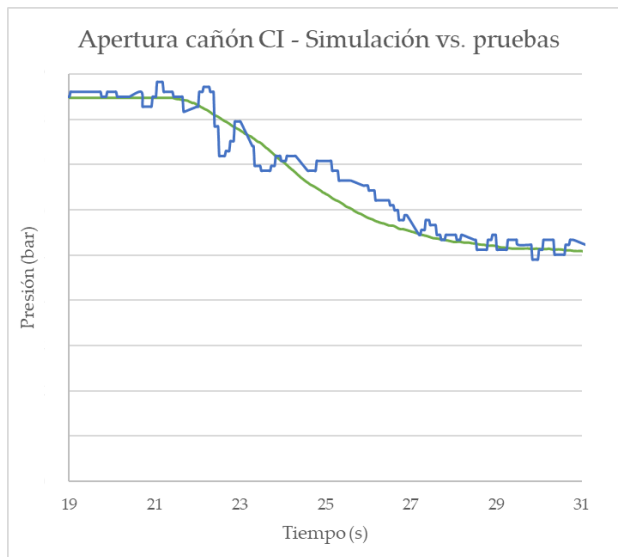


Imagen 5-14: Comparativa resultados (sensor PT007) real vs simulados (prueba3)

Estos análisis se completaron con el estudio de fenómenos en estado transitorio (apertura/cierre de consumidores o de válvulas) o también el arranque/parada de la bomba. En el caso concreto de la apertura del monitor contraincendios se pudieron ajustar los parámetros de apertura respecto a lo observado en las pruebas obteniendo lo que se puede observar en la imagen 5-15.



**Imagen 5-15: Comparativa de los resultados obtenidos para la caída de presión en un sensor tipo (real vs. simulado)**

### 5.6.5. Simulaciones realizadas para generación de datos

Como se ha comentado, dado que los resultados obtenidos de las pruebas no resultaron válidos para ser empleados como inputs de la red neuronal, se optó por emplear datos simulados. Es evidente que el uso de datos simulados para la simulación de roturas, que no han podido ser validados con datos reales, puede que no representen de la forma más ajustada posible el funcionamiento real del sistema, pero dado que el objetivo, en esta fase del trabajo era plantear la arquitectura necesaria para red neuronal (estructura, algoritmos de aprendizaje/entrenamiento, etc.) y a la espera de poder realizar nuevas pruebas a bordo para obtener valores reales válidos, se optó por emplear datos simulados.

En concreto, se modeló lo siguiente:

- Modelado de apertura de consumidores: Se generaron 22 eventos de funcionamiento normal del sistema en 17 ubicaciones diferentes (apertura de consumidores de agua salada, principalmente eyectores, de los que se conocen sus características de caudal y la velocidad de apertura).

**Tabla 5-2: Listado de aperturas de consumidores para caracterización de comportamiento normal (simulación)**

Ubicación	Elemento	Caudal
Ub -1	Eyector de Achique N°1	45 m3/h
Ub -2	Eyector de Achique N°2	45 m3/h
Ub -3	Eyector de Achique N°3	45 m3/h
Ub -4	Eyector de Achique N°4	45 m3/h
Ub -5	Eyector de Achique N°5	45 m3/h
Ub -6	Eyector de Achique N°6	45 m3/h
Ub -7	Eyector de Achique N°7	45 m3/h
Ub -8	Estación CI N°21	21.6 m3/h
Ub -9	Estación CI N°15	21.6 m3/h
Ub -10	Estación CI N°29	21.6 m3/h
Ub -11	Estación CI N°25	21.6 m3/h
Ub -12	Refr. Hélice Tranv. Pr.	5.4 m3/h
Ub -13	Estación espuma	155 m3/h
Ub -14	Rociado pañol cañón	91.2 m3/h
Ub -15	Cañón CI	400 m3/h
Ub -16	Cañón CI	400 m3/h
Ub -17	Planta AR	65 m3/h

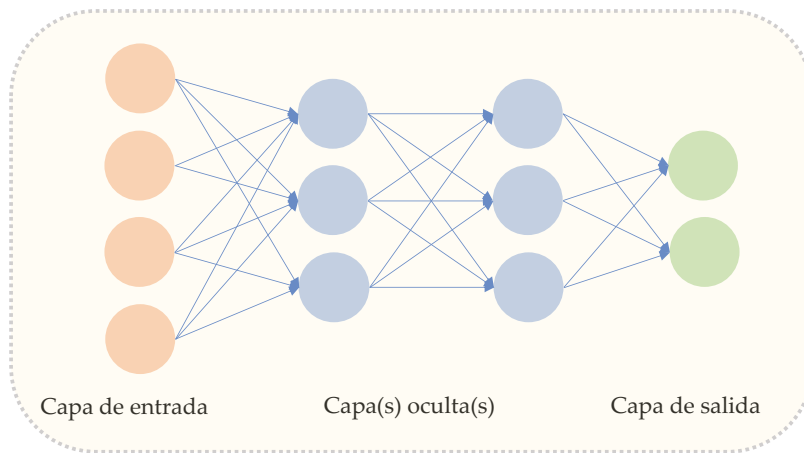
- Modelado de rotura de tuberías: Se llevaron a cabo la simulación de 35 casos de rotura de tuberías, repartidas de forma uniforme en el buque y modeladas según lo recomendado por los desarrolladores del software empleado. El resumen de los casos de rotura se incluye en la Tabla 5-3.

**Tabla 5-3: Ubicaciones de roturas de tuberías para caracterización de comportamiento en fallo (simulación)**

Ubicación	Elemento
Colector principal alto	4 roturas totales y 2 roturas parciales
Colector principal bajo	4 roturas totales y 2 roturas parciales
Ramales principales	10 roturas totales instantáneas y 10 parciales instantáneas
Ramales secundarios	7 roturas totales instantáneas y 6 parciales instantáneas

## 5.7. DESARROLLO DEL SISTEMA DE DETECCIÓN DE FALLOS

El algoritmo desarrollado, basado en redes neuronales, tiene un esquema general como el siguiente:



**Imagen 5-16: Esquema general para una red neuronal**

Las entradas de la red neuronal serán las series de datos correspondientes a las medidas de presión en cada uno de los sensores (tiempo, presión), mientras que la salida será un valor 0/1, es decir, fallo o funcionamiento normal del sistema.

### 5.7.1. Diseño de la red

#### 5.7.1.1. Estructura de los datos de entrada/salida

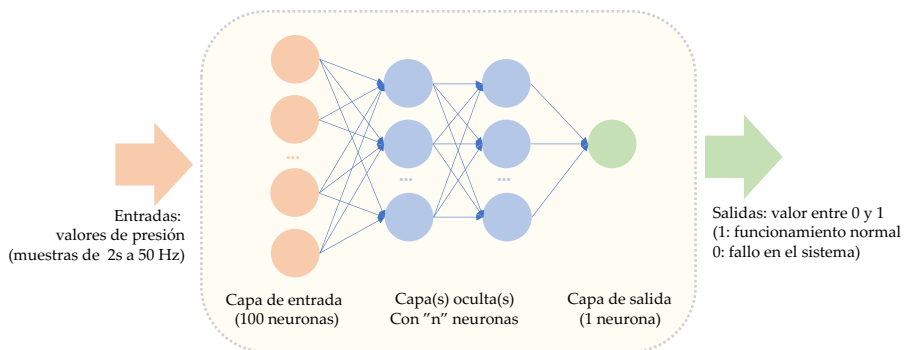
Se disponen de datos de las simulaciones en el sistema en los que se conocen los valores de la presión en los sensores y el evento correspondiente a esas mediciones (funcionamiento normal del sistema o rotura de un tramo de tubería). La estructura de los datos se corresponde con lo siguiente: Para cada simulación (se han realizado un total de 57 casos diferentes) se cuenta con un histórico de datos de los sensores correspondiente a 100 s de simulación (valores de presión a 50 Hz, 50 mediciones por segundo). En este histórico estará identificado el instante en el que tiene lugar un evento: apertura de consumidores de agua o rotura de tubería.

### 5.7.1.2. Selección del tamaño de las muestras para el entrenamiento

Inicialmente se seleccionó un intervalo de tiempo de estudio (columnas de la matriz de inputs) de 1s, lo que se correspondía con 50 mediciones. Se llevaron a cabo pruebas preliminares con redes neuronales y se comprobó que 1s no era tiempo suficiente para determinar si en ese espacio de tiempo se había llevado a cabo la apertura de un consumidor del sistema, o si lo que está sucediendo es la rotura de una tubería. Por ello, se pasó a seleccionar un intervalo de tiempo para las muestras de 2 segundos, con los que los resultados obtenidos mejoraron de forma importante. Se realizó una última prueba, empleando muestras de 3 segundos de duración, no aportando mejoras significativas a los resultados, por lo que se optó por mantener 2 segundos como el tamaño a considerar en esta fase del desarrollo.

### 5.7.1.3. Selección del tipo de red tipo de aprendizaje, entrenamiento y funciones de activación

Como ya se ha adelantado, se optó por seleccionar una red neuronal feed-forward fully-connected, dado que se trata de las estructuras de red neuronal más ampliamente utilizadas y versátiles. Una red de tipo feed-forward (o red prealimentada) se caracteriza porque es unidireccional (no cíclica), la información sigue una dirección de izquierda (capa de entrada) a derecha (capa de salida). Además, fully-connected implica que todas las neuronas de una capa están conectadas con todas las neuronas de la siguiente capa. En resumen, la red tendrá la siguiente estructura:



**Imagen 5-17: Esquema general de la red neuronal empleada (número de capas y neuronas por capa por definir)**

Como también se ha adelantado, los datos que se tienen para el entrenamiento de la red contienen tanto los inputs como los outputs, lo que va a permitir realizar un aprendizaje supervisado. Se empleará un tipo de aprendizaje por corrección de error, en concreto un algoritmo de tipo back-propagation/descenso del gradiente para que la red autoajuste sus parámetros. En cuanto a algoritmo de entrenamiento se seleccionó la Regularización Bayesiana, ya que con ella se evitan posibles problemas de sobreentrenamiento. Esta toma de decisión se realizó basándose en los resultados preliminares obtenidos con la herramienta de MATLAB, NFTOOL, la cual permite de forma muy sencilla, sin programar, entrenar una red neuronal, y mediante la cual, se pudo comprobar, que los resultados obtenidos con la regulación bayesiana mejoraban de forma importante los que se obtenían empleando Levenberg-Marquardt.

#### 5.7.1.4. Selección del número de capas y número de neuronas

La elección del número de capas y de neuronas a emplear en una red neuronal no tiene una solución directa. No hay ninguna norma establecida que indique cuántas capas o cuántas neuronas se deben emplear en el caso de una red neuronal determinada, aunque sí se pueden recopilar una serie de comentarios o recomendaciones:

- Una gran parte de los problemas prácticos se resuelven empleando una única capa oculta.
- El entrenamiento se ralentiza al aumentar el número de capas.
- Aumentar las capas ocultas hace que la capa oculta adicional a través de la cual se propaga el error hace el gradiente más inestable.
- Aumentar el número de capas ocultas incrementa el número de mínimos locales.
- El uso de muchas neuronas en la capa oculta puede generar problemas de sobreentrenamiento (la red particulariza, no generaliza).
- El uso de un número demasiado pequeño de neuronas puede causar subentrenamiento, con lo que la red no resuelve el problema.
- Existe una regla denominada “regla de la pirámide” según la cual el número de neuronas sigue una forma de pirámide,  $h = \sqrt{m \cdot n}$ , siendo: “n” el número de neuronas de entrada, “m” el número de neuronas de salida, y “h” el número de neuronas en la capa oculta.



De modo que se decidió optar por realizar un análisis de sensibilidad, de cara a obtener la estructura óptima (aquella en la que se minimiza el error)

- Número de capas: que se variarán de 1 a 3.
- Número de neuronas: que se variarán de 5 a 30.

#### **5.7.1.5. Selección de los casos de entrenamiento/test/validación**

En los primeros entrenamientos llevados a cabo se optó por seleccionar una selección de casos de tipo aleatorio. Después se complementó con entrenamientos realizados con la selección de casos para entrenamiento/test/validación realizada de forma manual.

#### **5.7.1.6. Inicialización de parámetros iniciales**

Una vez que se ha creado la red neuronal, para poder trabajar con ella, será necesario inicializarla, de este modo se inicializan los valores de la red: pesos y sesgo. Esto es especialmente importante en casos de reentrenamiento, ya que, si no se reinicia la red, los re-entrenamientos se realizarían empleando siempre los mismos valores de pesos y sesgos iniciales.

### **5.7.2. Entrenamiento de la red**

Con todos estos condicionantes, se llevó a cabo la programación de la red (en Matlab). En la imagen 5-18 se describe mediante un diagrama de flujo el programa desarrollado. Se realizaron, para cada  $n^{\circ}$  de capas y para cada  $n^{\circ}$  de neuronas: 50 iteraciones en las cuales se inicializan los valores de la red (pesos y bias), así como la división de datos para entrenamiento/validación/test. Cada una de esas 50 iteraciones se entrena 100 veces y se seleccionan las 3 mejores redes: la que tiene un menor error en entrenamiento, un menor error en validación y un menor error en test. Se almacenan las “mejores” redes de cada iteración, para comparación de los resultados obtenidos.

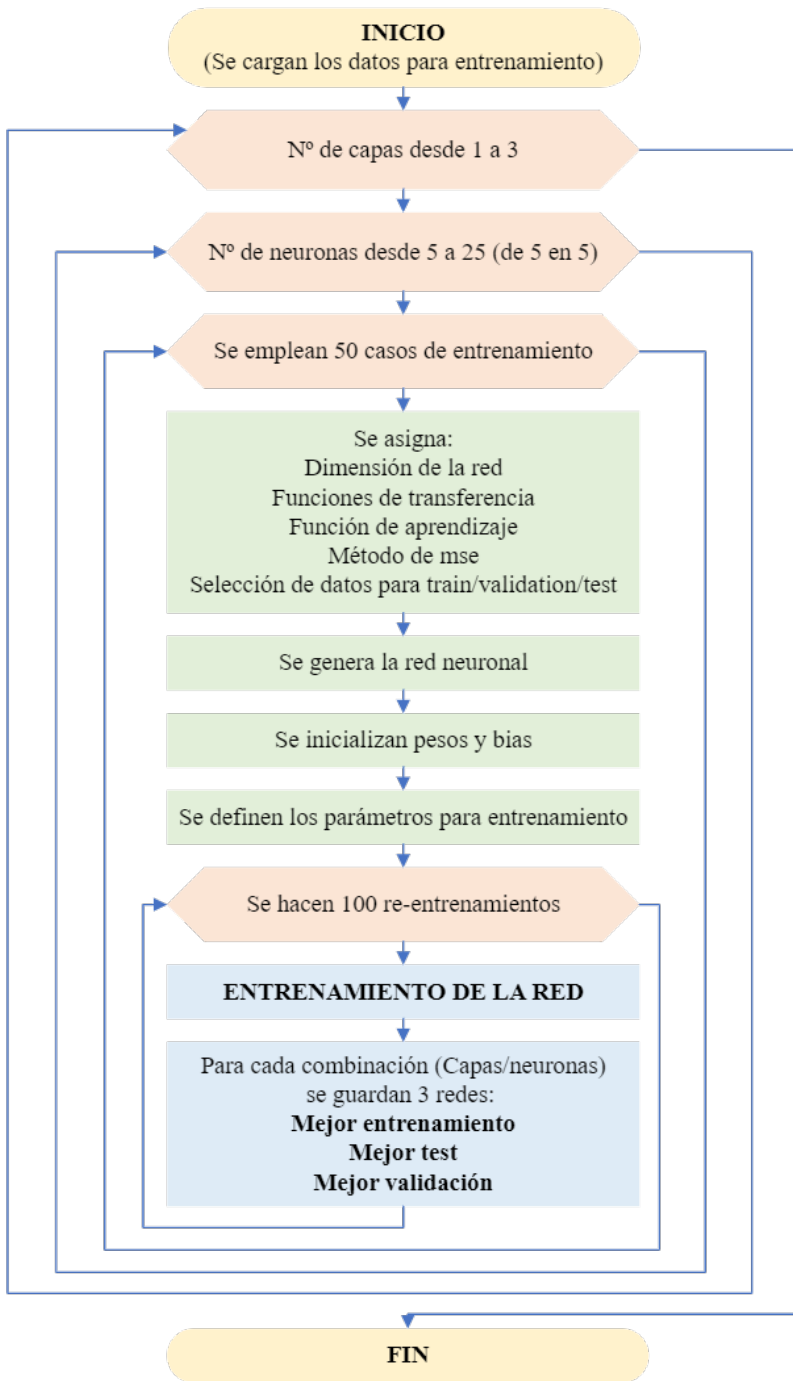


Imagen 5-18: Diagrama de flujo del código desarrollado para entrenamiento de la red para detección de fallos

### 5.7.3. Resultados

Como se ha comentado, la selección de casos para entrenamiento, validación y test se llevó a cabo de dos formas: por un lado, se realizó una selección manual de los casos y, por otro lado, se llevó a cabo una selección aleatoria. En el caso de la selección manual, se obtuvieron los siguientes resultados. La forma en la que se presentan los resultados en la siguiente tabla es la siguiente: para cada combinación de número de capas y número de neuronas, se presentan los resultados obtenidos en las 3 redes que guarda el programa:

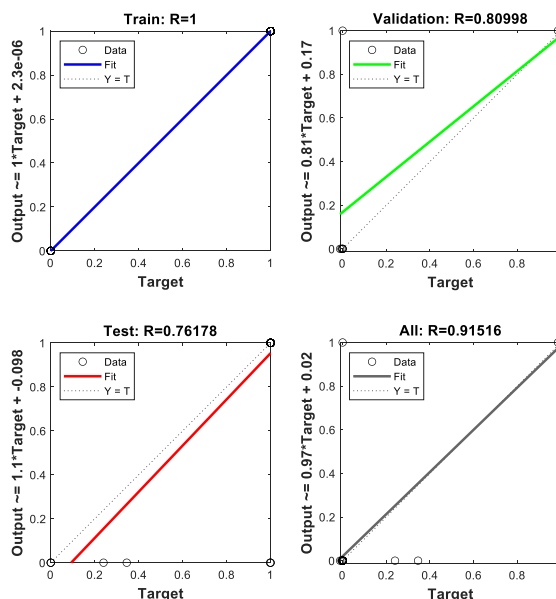
- BEST PERF: La red que presenta un menor error en entrenamiento.
- BEST VAL PERF: La red que presenta un menor error en validación.
- BEST TEST PERF: La red que presenta un menor error en test.

Para cada una de esas 3 redes almacenadas, se presentan los resultados concretos obtenidos para esa red (error en entrenamiento, error en validación y error en test):

**Tabla 5-4: Resumen de resultados obtenidos en el entrenamiento de la red neuronal (selección de casos entrenamiento/validación/test manual)**

		1 CAPA					
		5 NEURS	10 NEURS	15 NEURS	20 NEURS	25 NEURS	30 NEURS
BEST PERF	best_perf	4.208E-05	2.963E-10	3.691E-10	4.551E-11	2.705E-09	2.564E-12
	best_vperf	3.182E-02	4.992E-02	4.251E-02	4.167E-02	4.504E-02	6.360E-02
	best_tperf	4.531E-02	5.633E-02	5.531E-02	4.406E-02	4.176E-02	6.632E-02
BEST VAL	best_perf	1.244E-03	9.096E-03	9.406E-04	1.625E-02	4.905E-03	1.841E-03
	best_vperf	2.561E-02	2.212E-02	2.502E-02	1.529E-02	2.700E-02	2.634E-02
	best_tperf	5.243E-02	6.502E-02	5.080E-02	9.799E-02	6.708E-02	5.669E-02
BEST TEST	best_perf	5.228E-04	5.489E-03	2.028E-02	7.006E-04	1.507E-02	1.514E-02
	best_vperf	3.184E-02	3.913E-02	4.532E-02	4.587E-02	6.785E-02	4.276E-02
	best_tperf	3.633E-02	3.644E-02	3.434E-02	2.809E-02	2.538E-02	4.338E-02
		2 CAPAS					
		5 NEURS	10 NEURS	15 NEURS	20 NEURS	25 NEURS	30 NEURS
BEST PERF	best_perf	9.887E-06	5.304E-04	3.679E-04	7.595E-04	2.337E-04	4.634E-07
	best_vperf	3.465E-02	2.498E-02	2.887E-02	2.983E-02	3.182E-02	2.546E-02
	best_tperf	1.047E-01	4.995E-02	4.977E-02	5.256E-02	4.967E-02	3.264E-01
BEST VAL	best_perf	6.218E-03	5.751E-03	6.008E-03	7.112E-03	6.882E-03	9.645E-03
	best_vperf	2.169E-02	2.245E-02	2.347E-02	2.212E-02	2.385E-02	2.449E-02
	best_tperf	1.003E-01	8.160E-02	8.307E-02	1.029E-01	1.118E-01	4.802E-02
BEST TEST PERF	best_perf	2.077E-02	7.052E-04	9.326E-04	1.675E-02	5.211E-04	7.265E-04
	best_vperf	4.569E-02	3.322E-02	2.922E-02	3.800E-02	2.869E-02	3.061E-02
	best_tperf	3.973E-02	4.624E-02	4.555E-02	4.569E-02	4.485E-02	4.683E-02
		3 CAPAS					
		5 NEURS	10 NEURS	15 NEURS	20 NEURS	25 NEURS	30 NEURS
BEST PERF	best_perf	4.077E-03	7.576E-04	3.427E-03	3.556E-03	1.581E-03	2.801E-03
	best_vperf	1.591E-02	3.411E-02	2.484E-02	1.673E-02	1.974E-02	1.676E-02
	best_tperf	4.767E-02	4.767E-01	5.122E-02	1.315E-01	4.451E-02	4.584E-02
BEST VAL	best_perf	5.191E-03	2.754E-03	3.586E-03	5.055E-03	3.581E-03	3.974E-03
	best_vperf	1.292E-02	1.435E-02	1.474E-02	1.426E-02	1.305E-02	1.152E-02
	best_tperf	7.405E-02	2.676E-02	1.016E-01	5.709E-02	4.729E-02	5.035E-02
BEST TEST	best_perf	2.570E-02	2.754E-03	2.494E-02	2.584E-02	1.581E-03	3.699E-03
	best_vperf	4.876E-02	1.435E-02	4.784E-02	4.845E-02	1.974E-02	1.168E-02
	best_tperf	4.140E-02	2.676E-02	4.029E-02	4.994E-02	1.974E-02	4.422E-02

Los menores errores en entrenamiento se obtienen para redes de una sola capa. Para los errores obtenidos en validación o en test, las diferencias no son significativas. Se ha de destacar que los resultados obtenidos han sido relativamente buenos, ya que en general, no hay muchos casos que se escapen del ajuste. Y, en todo caso, parece razonable que exista alguno, ya que, en la simulación de casos realizada, existen aperturas normales y roturas que tienen un comportamiento no muy alejado (aperturas muy rápidas y roturas no instantáneas). En la imagen 5-19, se presenta, a modo de ejemplo, el resultado obtenido para una red de 1 capa con 20 neuronas (gráficas de regresión para entrenamiento, test, validación y total).



**Imagen 5-19: Regresión obtenida con 1 capa y 20 neuronas**

Si bien, se ha de señalar que el principal objetivo en esta etapa del trabajo no recae en obtener resultados ajustados, sino en validar la metodología propuesta para definición de la red neuronal y su entrenamiento, si se busca aplicar un sistema de detección de fallos como el descrito para ser aplicado a un sistema real.

En el caso de la selección aleatoria de los casos empleados para entrenamiento, validación y test, se obtuvieron los siguientes resultados para cada combinación de  $N^{\circ}$  capas/ $N^{\circ}$  neuronas:

**Tabla 5-5: Resumen de resultados obtenidos en el entrenamiento de la red neuronal (selección de casos entrenamiento/validación/test aleatoria)**

		1 CAPA					
		5 NEURS	10 NEURS	15 NEURS	20 NEURS	25 NEURS	30 NEURS
BEST PERF	best_perf	2.361E-13	2.703E-12	9.767E-13	3.353E-13	5.351E-13	2.470E-13
	best_vperf	3.597E-09	3.620E-05	7.959E-05	1.092E-10	1.961E-02	5.340E-10
	best_tperf	1.961E-02	1.904E-02	2.230E-02	2.333E-02	4.534E-02	3.022E-02
BEST VAL PERF	best_perf	1.682E-11	4.610E-12	3.589E-03	1.572E-07	4.172E-03	6.422E-03
	best_vperf	2.916E-13	2.142E-12	2.571E-12	3.694E-13	2.155E-13	1.070E-13
	best_tperf	1.961E-02	1.491E-03	5.061E-03	8.339E-13	1.321E-07	2.083E-13
BEST TEST PERF	best_perf	4.167E-03	4.053E-03	4.738E-03	3.079E-02	4.167E-03	6.422E-03
	best_vperf	9.384E-12	6.116E-07	7.959E-05	9.065E-13	2.808E-10	1.070E-13
	best_tperf	1.164E-12	2.075E-12	1.429E-12	1.101E-13	1.421E-13	2.083E-13

		2 CAPAS					
		5 NEURS	10 NEURS	15 NEURS	20 NEURS	25 NEURS	30 NEURS
BEST PERF	best_perf	4.931E-13	6.494E-11	9.310E-11	2.096E-11	1.631E-10	2.203E-11
	best_vperf	1.120E-08	1.856E-05	1.013E-08	1.696E-09	7.375E-11	2.100E-07
	best_tperf	6.624E-03	9.032E-02	2.300E-02	1.421E-01	3.937E-02	7.348E-02
BEST VAL	best_perf	1.859E-05	1.920E-02	8.953E-03	9.098E-03	1.631E-10	4.161E-03
	best_vperf	1.332E-13	1.920E-02	4.580E-11	5.303E-11	7.375E-11	1.566E-11
	best_tperf	6.536E-03	2.388E-08	4.695E-05	2.404E-10	3.937E-02	2.353E-08
BEST TEST	best_perf	1.402E-03	1.794E-02	4.888E-03	9.098E-03	1.275E-02	1.561E-02
	best_vperf	2.405E-05	1.294E-02	2.474E-10	5.637E-09	1.961E-02	9.571E-06
	best_tperf	2.526E-13	6.034E-11	4.111E-11	4.838E-11	5.163E-11	6.658E-12

		3 CAPAS					
		5 NEURS	10 NEURS	15 NEURS	20 NEURS	25 NEURS	30 NEURS
BEST PERF	best_perf	1.702E-10	1.114E-09	2.376E-10	2.249E-10	3.288E-10	9.982E-10
	best_vperf	2.504E-08	1.206E-07	2.469E-06	1.560E-09	2.399E-08	1.841E-08
	best_tperf	2.155E-02	3.348E-04	1.491E-01	7.176E-02	3.788E-02	1.321E-02
BEST VAL PERF	best_perf	1.132E-02	7.117E-05	1.250E-02	1.525E-02	4.509E-03	2.807E-03
	best_vperf	4.144E-11	6.922E-10	2.190E-09	1.115E-10	1.838E-10	5.368E-10
	best_tperf	9.726E-04	1.880E-08	4.498E-09	3.869E-10	3.906E-04	6.974E-09
BEST TEST PERF	best_perf	1.153E-02	3.422E-03	4.165E-03	1.510E-02	8.049E-03	4.508E-03
	best_vperf	1.153E-02	1.960E-02	3.921E-02	7.002E-04	2.647E-09	7.772E-03
	best_tperf	5.086E-11	6.230E-10	1.671E-09	1.191E-10	3.121E-10	7.475E-10

Los resultados obtenidos para una selección aleatoria de los casos de entrenamiento, validación y test, son significativamente mejores que en el caso de la selección manual. De nuevo, se incluye a modo de ejemplo, los resultados obtenidos (representación gráfica de la regresión realizada) para 1 capa y 15 neuronas (imagen 5-20):

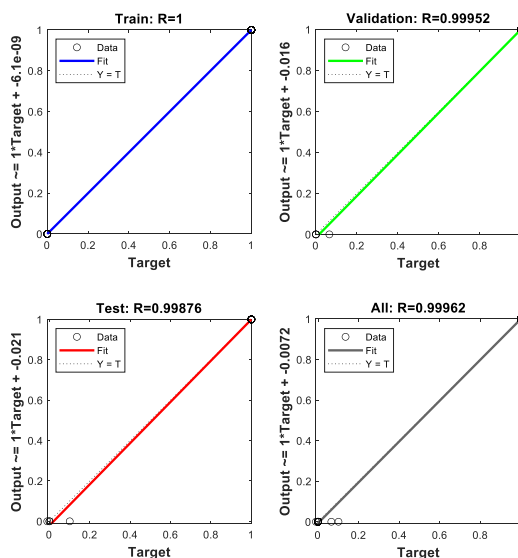


Imagen 5-20: Regresión para 1 capa oculta y 15 neuronas

Como se puede deducir de los resultados obtenidos, no es necesario ir a una red de gran tamaño para obtener resultados satisfactorios. Aunque, se esperará a emplear datos reales para hacer conclusiones respecto a la estructura de la red neuronal.

## 5.8. PRUEBAS EN BUQUE REAL. FASE 2

Con el objetivo de completar todo el trabajo anterior, y dado que los resultados obtenidos en las pruebas de la fase 1 fueron poco aprovechables, se propuso la realización de nuevas pruebas a bordo de un buque. La idea inicial era tratar de realizar pruebas en el mismo buque que se empleó en la primera fase. Finalmente, no se pudo disponer de ese buque, por lo que se decidió realizar las pruebas a bordo de otro buque, en este caso, un buque en operación de la Armada Española.

Se llevó a cabo una propuesta para la realización de estas nuevas pruebas a bordo. Para ello se analizó el sistema CI del nuevo buque: elementos que los componen (bombas, sensores, consumidores), localización en el buque de la red de sensores de presión y de los consumidores. El buque cuenta con 6 bombas CI de caudal nominal: 225 m<sup>3</sup>/h y 9 bar. Se disponen a lo largo del buque 35 sensores de presión a lo largo del sistema contraincendios por agua salada.

### 5.8.1. Preparación de las pruebas

Se elaboró una nueva propuesta para la elaboración de las pruebas y que se definió de forma análoga a lo realizado en las pruebas de la primera fase.

La preparación de las pruebas se hizo agrupando las diferentes pruebas se agruparon siguiendo la siguiente estructura:

- Prueba N<sup>o</sup>1 - Caracterización del arranque y parada de bombas del sistema. Con el objetivo de caracterizar las curvas de presión correspondientes a ese evento.
- Prueba N<sup>o</sup>2 - Caracterización de apertura y cierre de consumidores: El objetivo de esta prueba fue caracterizar las curvas de presión correspondientes a la apertura/cierre de consumidores del sistema. En concreto, se propuso lo siguiente:
  - o Apertura (y cierre) de los eyectores disponibles.
  - o Apertura (y cierre) de boquiles contraincendios. Se seleccionó al menos 1 boquil en cada ubicación: cubierta principal – popa, cubierta principal – proa, nivel 01 y nivel 02.
  - o Apertura (y cierre) del servicio de descontaminación popa/proa.
  - o Apertura (y cierre) del sistema de rociado de pañoles.
  - o Apertura (y cierre) del circuito de emergencia de los diésel-generadores (DDGG).
- Prueba N<sup>o</sup> 3 - Simulación de rotura de tubería: Para las pruebas de rotura y colapso de tubería se requirió (al igual que en la primera fase) de una preparación previa consistente en la colocación de una válvula de ¼ de vuelta y manguera en un boquil de 2 ½ para permitir un cierre parcial y apertura casi instantáneo. Para la simulación de la rotura de tubería se realizó esa apertura rápida en las mismas ubicaciones que



las aperturas y cierres correspondientes a la caracterización de los boquiles contraincendios de la prueba N°2.

- Prueba N° 4 - Simulación de colapso de tubería. Para la simulación del colapso de tubería se realizaron cierres parciales casi instantáneos en las mismas ubicaciones que las que se consideraron en la prueba N°3.
- Prueba N° 5 - Caracterización de caída de presión en sistema: La última prueba consistió en la apertura consecutiva de los eyectores de achique disponibles, hasta que se produzca una notable caída de presión que requiera el arranque de una bomba adicional en el sistema contraincendios. A continuación, se llevaría a cabo el cierre de los eyectores, en el orden inverso al de la apertura y la parada de la bomba que se había arrancado de forma adicional.

### **5.8.2. Realización de pruebas**

En primer lugar, se llevaron a cabo pruebas preliminares en el buque, con el fin de comprobar que el sistema para recogida de datos desarrollado por Navantia recogía datos de la red del sistema de control del buque y además de ello, comprobar que era posible modificar la programación de las LSS para recoger los datos de los sensores de presión con una precisión de la milésima de bar. Las pruebas fueron parcialmente satisfactorias: se recogieron datos y la precisión (en la LSS-FZ2/1 modificada) fue la deseada. No sucedió lo mismo en el caso de la frecuencia de grabado de las señales. A pesar de que los datos se solicitaban a los PLCs cada 20 ms, debido a la frecuencia máxima permitida por los PLCs los datos con menos espaciado que se pudieron recoger se obtuvieron con una diferencia de 100 ms. Pese a ello, y como se verá más adelante, esa frecuencia de muestreo fue suficiente para el reentrenamiento de la red neuronal de detección de fallos y para la validación del modelo empleado para la localización de fallos.

Finalmente, el 10 de mayo de 2021 se realizaron las pruebas definitivas en el buque, siguiendo el procedimiento acordado. En las siguientes imágenes (5-21 a 5-24) se adjuntan algunos resultados obtenidos:

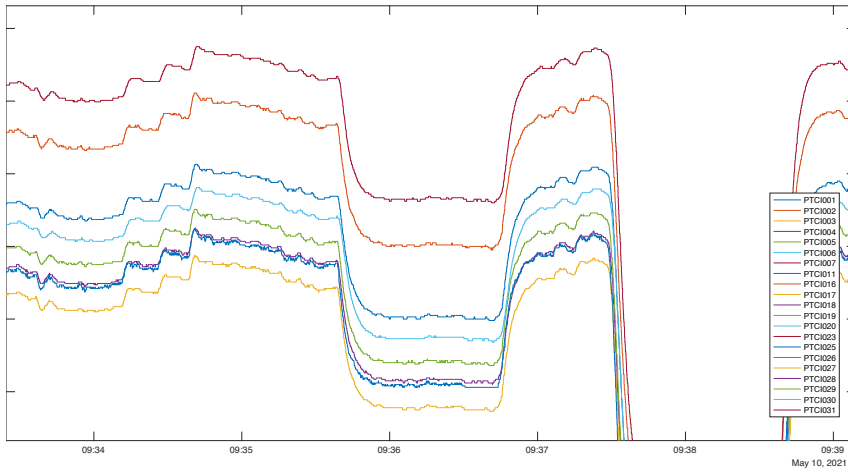


Imagen 5-21: Caracterización apertura de eyectores

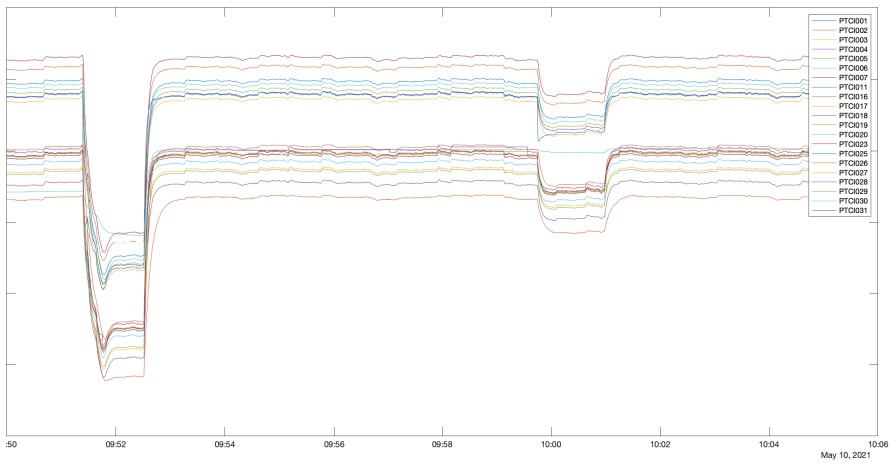


Imagen 5-22: Caracterización apertura rociadores

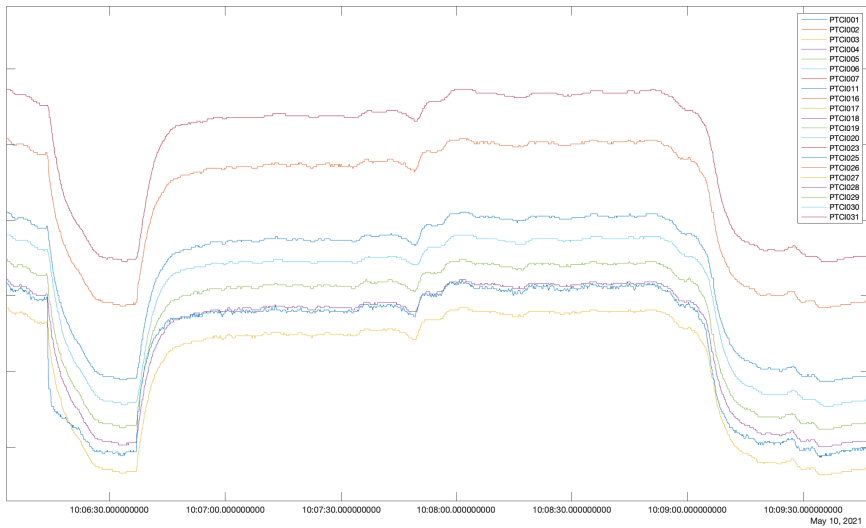


Imagen 5-23: "simulación de rotura" en boquil CI y apertura normal de un boquil contraincendios

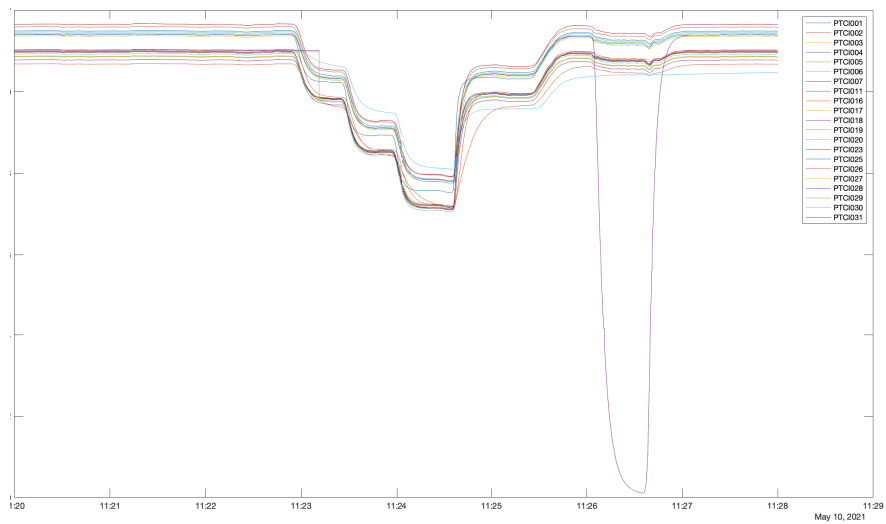


Imagen 5-24: Apertura eyectores. Parada/arranque bomba CI

### 5.8.3. Conclusiones/resumen de las pruebas realizadas

Estas pruebas correspondientes a la fase 2, se llevaron a cabo en un buque de más de 20 años de antigüedad. Para poder realizar la adquisición de datos fue necesario realizar modificaciones en las unidades intermedias de adquisición de datos desde los sensores hacia el sistema de control del buque (para poder disponer de los datos de presión con una precisión de décima de bar y a la mayor frecuencia de refresco posible). Esta modificación implicó reprogramar una a una estas unidades intermedias, demorando el inicio de las pruebas. Además, se presentaron problemas en la adquisición de datos a través del registrador (datalogger), por lo que fue necesario modificar el punto de conexión a la red del sistema de control en la cámara de control. Este motivo también provocó demoras en el inicio de las pruebas, por lo que finalmente el tiempo disponible, se vio reducido a la mitad.

En cuanto a los datos finalmente recogidos:

- Se consiguió recoger datos de 27 sensores, todos ellos con una resolución de milésima de bar.
- La frecuencia de refresco de las señales fue, como máximo de 10 Hz. Se ha considerado válida para esta etapa del estudio, pero se recomienda una frecuencia superior a 20 Hz (e ideal de 50 Hz).
- Se guarda un dato de presión siempre que este cambie de valor, y esto es independiente para cada uno de los sensores. Es un aspecto importante que se ha de tener en cuenta, ya que la red neuronal lee únicamente vectores que contienen una serie de datos de presión. Dichos vectores deben de tener un equi-espaciado entre valores constante. En el caso del uso de los valores offline, esto no implica ningún tipo de problema, ya que se puede realizar una interpolación, pero sí tendrá relevancia en el caso de que se esté trabajando online. En las pruebas realizadas en la primera fase, se guardaron datos a 50 Hz, independientemente del cambio de valor.

## **5.9. REDISEÑO/REENTRENAMIENTO DEL SISTEMA DE DETECCIÓN DE FALLOS. DESARROLLO DEL SISTEMA DE LOCALIZACIÓN DE FALLOS**

### **5.9.1. Rediseño y reentrenamiento de la red neuronal para detección de fallos**

Con los datos recogidos en las pruebas realizadas a bordo en esta segunda fase, se llevó a cabo el reentrenamiento de la red neuronal para detección de fallos. Se partió de la red desarrollada en la primera fase, y con el objetivo inicial de validar su uso, o minimizar las modificaciones en ella. En el entrenamiento realizado con datos simulados, el tamaño de los casos para entrenamiento fue de 100 elementos, correspondiente a 2 segundos de tiempo, pero después de analizar los resultados obtenidos en las pruebas realizadas, se planteó emplear 3 tamaños diferentes para las muestras:

- Un tamaño mínimo de 20 elementos (20 valores de presión), correspondiente a 3.5 segundos.
- Un tamaño medio de 35 elementos, correspondiente a 6.5 segundos.
- Un tamaño máximo de 50 elementos, correspondiente a 9 segundos.

Se planteó un primer análisis para 1, 2 y 3 capas ocultas, y con un número de neuronas para cada capa variando entre: 5, 10, 20, 30 y 50 neuronas. Se muestran los resultados obtenidos para un tamaño de las muestras de 3.5 segundos (se procedió de la misma forma para tamaños de 6.5 segundos y 9 segundos):

**Tabla 5-6: Resultados obtenidos en el entrenamiento de la red neuronal (tamaño de la muestra de 3.5 s y para 1,2,3 capas)**

		1 CAPAS				
		5 NEURS	10 NEURS	20 NEURS	30 NEURS	50 NEURS
BEST PER-	<b>best_perf</b>	9.574E-02	9.876E-02	9.298E-02	1.031E-01	1.185E-01
	best_vperf	1.607E-01	1.181E-01	1.582E-01	1.330E-01	1.546E-01
	best_tperf	1.166E-01	1.611E-01	2.045E-01	1.602E-01	2.117E-01
BEST VALI-	<b>best_perf</b>	1.183E-01	1.065E-01	1.321E-01	1.433E-01	1.491E-01
	best_vperf	4.175E-02	6.326E-02	6.827E-02	6.556E-02	8.274E-02
	best_tperf	1.346E-01	2.007E-01	1.192E-01	8.233E-02	1.669E-01
BEST TEST	<b>best_perf</b>	1.324E-01	1.600E-01	1.273E-01	1.356E-01	1.530E-01
	best_vperf	1.175E-01	1.411E-01	1.490E-01	1.294E-01	1.497E-01
	best_tperf	5.657E-02	6.032E-02	5.671E-02	4.511E-02	9.935E-02

		2 CAPAS				
		5 NEURS	10 NEURS	20 NEURS	30 NEURS	50 NEURS
BEST PER-	<b>best_perf</b>	3.901E-02	5.092E-02	4.817E-02	4.513E-02	5.014E-02
	best_vperf	9.128E-02	9.343E-02	8.344E-02	8.411E-02	8.559E-02
	best_tperf	1.730E-01	2.105E-01	1.584E-01	1.075E-01	2.445E-01
BEST VALI-	<b>best_perf</b>	7.272E-02	9.587E-02	7.363E-02	7.714E-02	7.687E-02
	best_vperf	1.067E-02	2.360E-02	2.150E-02	1.508E-02	2.280E-02
	best_tperf	8.272E-02	9.730E-02	1.024E-01	5.767E-02	1.147E-01
BEST TEST	<b>best_perf</b>	9.433E-02	8.899E-02	6.908E-02	8.638E-02	8.054E-02
	best_vperf	1.003E-01	1.148E-01	7.558E-02	5.160E-02	8.573E-02
	best_tperf	1.067E-02	2.029E-02	2.137E-02	2.613E-02	2.306E-02

		3 CAPAS				
		5 NEURS	10 NEURS	20 NEURS	30 NEURS	50 NEURS
BEST PER-	<b>best_perf</b>	1.662E-02	2.172E-02	2.106E-02	2.583E-02	8.464E-03
	best_vperf	5.116E-02	1.367E-01	2.348E-01	9.912E-02	1.813E-01
	best_tperf	1.052E-01	1.866E-01	1.796E-01	1.788E-01	7.650E-01
BEST VALI-	<b>best_perf</b>	4.447E-02	8.563E-02	6.669E-02	6.625E-02	6.097E-02
	best_vperf	2.714E-03	6.542E-03	1.823E-02	2.241E-02	1.617E-02
	best_tperf	1.682E-02	3.207E-02	2.986E-02	4.874E-02	3.851E-02
BEST TEST	<b>best_perf</b>	4.106E-02	6.042E-02	6.543E-02	5.452E-02	6.215E-02
	best_vperf	3.974E-02	1.083E-01	6.181E-02	5.422E-02	3.058E-02
	best_tperf	3.627E-03	1.422E-02	1.323E-02	2.234E-02	1.862E-02

Como se puede observar, el error se reduce a medida que se amplía el número de capas ocultas. Además, parece que aumentar el número de neuronas no mejora los resultados obtenidos. Es algo que se observó para los 3 tamaños de muestra analizados (3.5, 6.5 y 9 segundos). Además, si se compara el error a igualdad de número de capas y neuronas y variando únicamente el tamaño de las muestras para entrenamiento se observa que aumentando el tamaño de la muestra se mejora la respuesta de la red (disminuye el error).

Con el objetivo de verificar lo observado, se amplió el estudio a 4, 5 y 6 capas. Analizando esos nuevos resultados parciales se pudo comprobar que aumentar el número de neuronas por capa por encima de 10 no hace disminuir el error (lo cual derivó en ampliar el estudio hasta 10 capas, pero únicamente para 5 y 10 neuronas) y que, en general, aumentar el tamaño de las muestras sí mejora el comportamiento de la red. En lo referente a los resultados por número de capas empleado, los mejores resultados se obtienen para configuraciones de entre 3 y 7 capas, pero no se puede concluir que aumentar o disminuir el número de capas mejore el resultado obtenido.

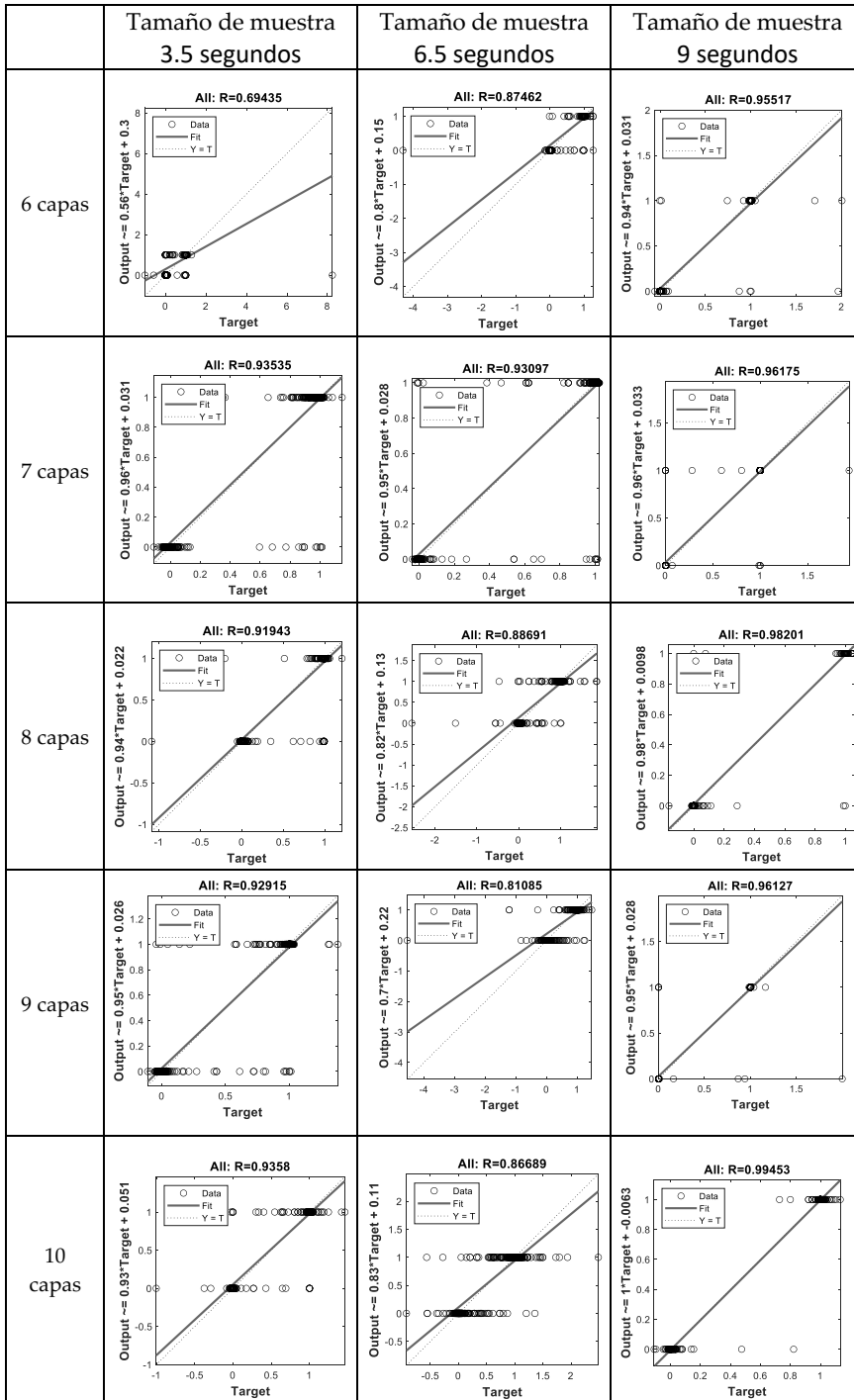
Para poder ilustrar de forma gráfica lo que se acaba de afirmar, se adjuntan en la siguiente tabla algunos de los resultados obtenidos en las regresiones correspondientes a las diferentes configuraciones de la red. A la hora de interpretar las gráficas que se presentan en la tabla se ha de tener en cuenta lo siguiente:

- Se presentan los resultados obtenidos para los 3 tamaños de muestra considerados, agrupados por columnas.
- En cada fila se varía el número de capas empleado en ese caso de estudio.
- Solo se presentan los resultados obtenidos para 5 neuronas (podría hacerse lo mismo para 10 neuronas, obteniendo una tabla similar).
- En la tabla solo se incluye (para simplificar) la gráfica que agrupa los resultados obtenidos para entrenamiento, test y validación (y no las gráficas de cada uno de los casos desagregados).

Tabla 5-7: Regresiones obtenidas en los entrenamientos de la red neuronal

	Tamaño de muestra 3.5 segundos	Tamaño de muestra 6.5 segundos	Tamaño de muestra 9 segundos
1 capa	<p>All: R=0.71309</p> <p>Output <math>\approx 1^{\circ}\text{Target} + 0.013</math></p>	<p>All: R=0.71171</p> <p>Output <math>\approx 0.92^{\circ}\text{Target} + 0.03</math></p>	<p>All: R=0.88842</p> <p>Output <math>\approx 0.93^{\circ}\text{Target} + 0.027</math></p>
2 capas	<p>All: R=0.83695</p> <p>Output <math>\approx 0.93^{\circ}\text{Target} + 0.05</math></p>	<p>All: R=0.85295</p> <p>Output <math>\approx 0.93^{\circ}\text{Target} + 0.048</math></p>	<p>All: R=0.98691</p> <p>Output <math>\approx 1^{\circ}\text{Target} + 0.0048</math></p>
3 capas	<p>All: R=0.91816</p> <p>Output <math>\approx 0.95^{\circ}\text{Target} + 0.034</math></p>	<p>All: R=0.9031</p> <p>Output <math>\approx 0.93^{\circ}\text{Target} + 0.058</math></p>	<p>All: R=0.98691</p> <p>Output <math>\approx 1^{\circ}\text{Target} + 0.0048</math></p>
4 capas	<p>All: R=0.84729</p> <p>Output <math>\approx 0.98^{\circ}\text{Target} + 0.089</math></p>	<p>All: R=0.93729</p> <p>Output <math>\approx 0.95^{\circ}\text{Target} + 0.032</math></p>	<p>All: R=0.97191</p> <p>Output <math>\approx 0.98^{\circ}\text{Target} + 0.011</math></p>
5 capas	<p>All: R=0.93178</p> <p>Output <math>\approx 0.94^{\circ}\text{Target} + 0.034</math></p>	<p>All: R=0.94023</p> <p>Output <math>\approx 0.93^{\circ}\text{Target} + 0.058</math></p>	<p>All: R=0.95471</p> <p>Output <math>\approx 0.95^{\circ}\text{Target} + 0.033</math></p>





Las principales conclusiones del entrenamiento de la red neuronal para detección de fallos en tuberías a partir de las mediciones de los datos de los sensores de presión del sistema CI por agua salada se pueden resumir en lo siguiente:

- En primer lugar, se ha de tener en cuenta que el entrenamiento de la red se ha hecho con un número bastante limitado de casos de funcionamiento normal y de rotura. Alcanzar un mayor ajuste pasará por poder entrenar a la red neuronal durante largos periodos y en condiciones reales para funcionamiento normal, así como en tratar de simular circunstancias anómalas (rotura o colapso) etiquetando esos datos como tal, para que el sistema ajuste mejor su comportamiento.
- Del análisis de la tabla 5-15, la conclusión más evidente es que el uso del tamaño de muestra máximo (de 9 segundos) es el que resulta en un mejor ajuste de la red.
- Se han analizado redes con un número de neuronas por capa comprendido entre 5 y 50 neuronas. Emplear neuronas por encima de 10 no hace que se mejore el comportamiento de la red.
- Se han analizado redes con un número de capas desde 1 a 10, obteniendo, en general, mejores resultados para un número de capas intermedio, no habiéndose observado que el uso de un número mínimo o máximo de capas mejore el comportamiento de la red.
- Para los diferentes tamaños de muestra, existe una o varias combinaciones de número de capas y número de neuronas con el cual se minimiza el error obtenido en el entrenamiento. En las imágenes 5-25, 5-26 y 5-27 se muestran a modo de ejemplo, alguna de las combinaciones que presentan un mejor ajuste:

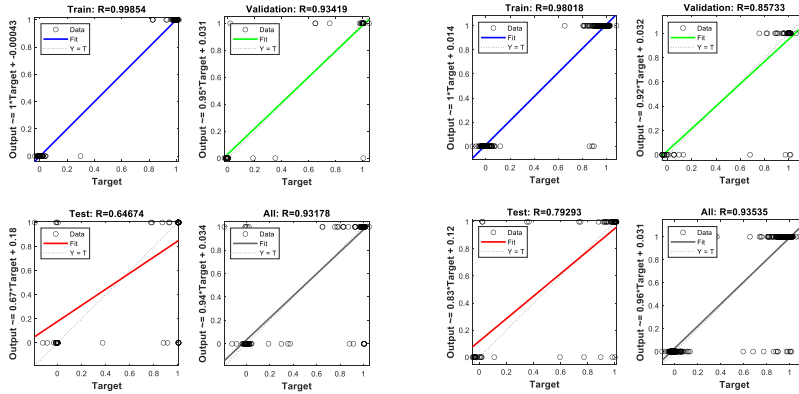


Imagen 5-25: Resultados obtenidos para 5 capas y 5 neuronas y para 7 capas con 5 neuronas (tamaño de muestra: 20 elementos)

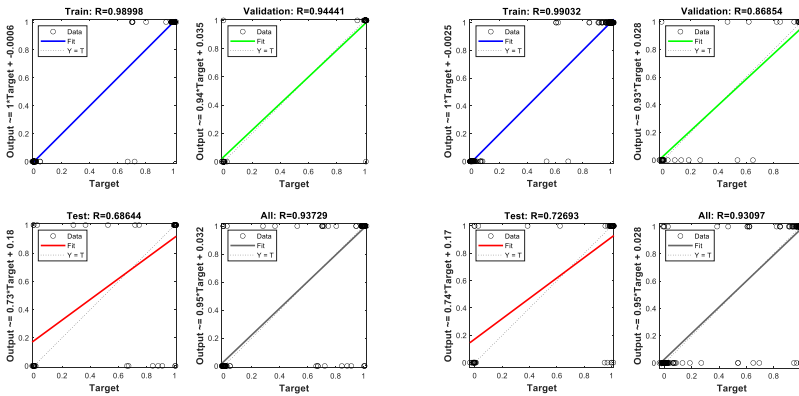


Imagen 5-26: Resultados obtenidos para 4 capas y 5 neuronas y para 7 capas con 5 neuronas (tamaño de muestra: 35 elementos)

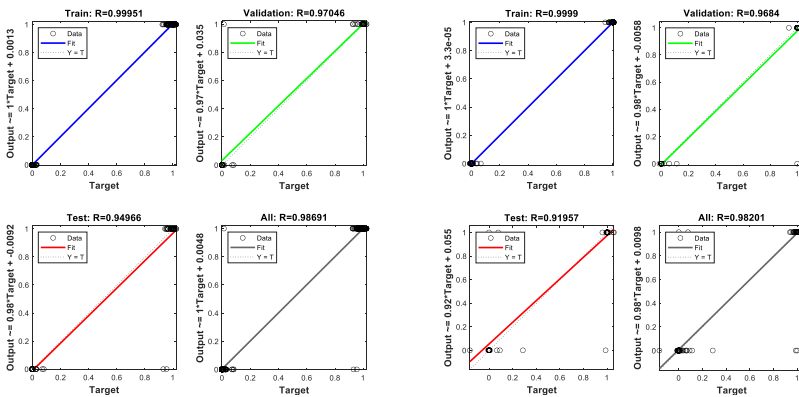
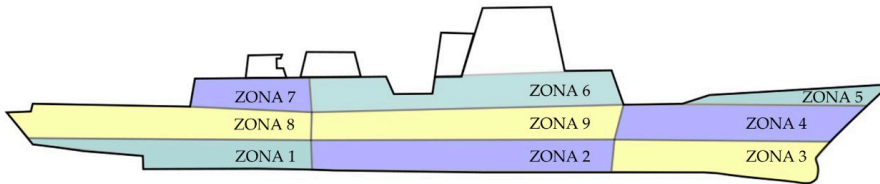


Imagen 5-27: Resultados obtenidos para 2 capas y 5 neuronas y para 8 capas con 5 neuronas (tamaño de muestra: 50 elementos)

### 5.9.2. Desarrollo de sistema para localización de fallos

A partir de los datos reales recogidos en las pruebas, se trató de probar una metodología que permitiese localizar de forma aproximada la ubicación del fallo. De nuevo se optó por el uso de redes neuronales, en este caso, a partir de las medidas de presión de los sensores, se determinan los retrasos de las señales. A partir de estos patrones de caídas de presión se trata de identificar de forma aproximada la localización del tramo del sistema en el que ha tenido lugar ese problema. Para ello, fue necesario realizar una **caracterización de la ubicación de los sensores / consumidores del sistema** para realizar el etiquetado de posicionado de elementos, que permita el entrenamiento del sistema basado en IA para la localización de fallos. Debido a la limitación en las pruebas realizadas, solo se han podido recoger datos de consumidores/fallos en determinadas posiciones del buque. Se dividió el buque en zonas, en las que se etiquetaron las diferentes pruebas realizadas, de acuerdo con lo que se muestra en la imagen 5-28.



**Imagen 5-28: Propuesta de división en zonas del buque para etiquetado de localización de fallos**

Para tratar de determinar el orden de caída de presión en los sensores y cómo relacionarlos con la ubicación de la apertura de consumidor/rotura del sistema CI se determinaron, para cada uno de los eventos, las caídas de presión en cada sensor para un tiempo determinado, en este caso, se determinaron las caídas de presión en cada segundo, obteniendo unas tablas para cada uno de los sensores y para cada uno de los eventos, con datos de caída de presión/segundo. Se muestran en las siguientes imágenes las caídas relativas de presión en cada uno de los sensores, para 4 eventos diferentes (2 de funcionamiento normal del sistema, y 2 de rotura de un tramo de tubería).

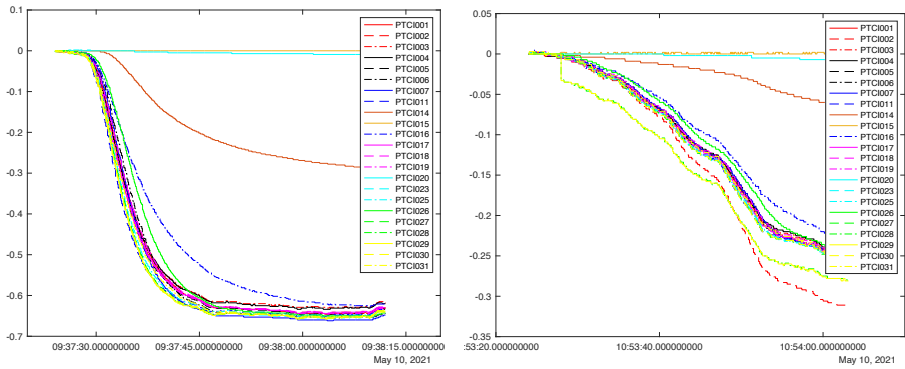


Imagen 5-29: Caídas de presión en funcionamiento normal

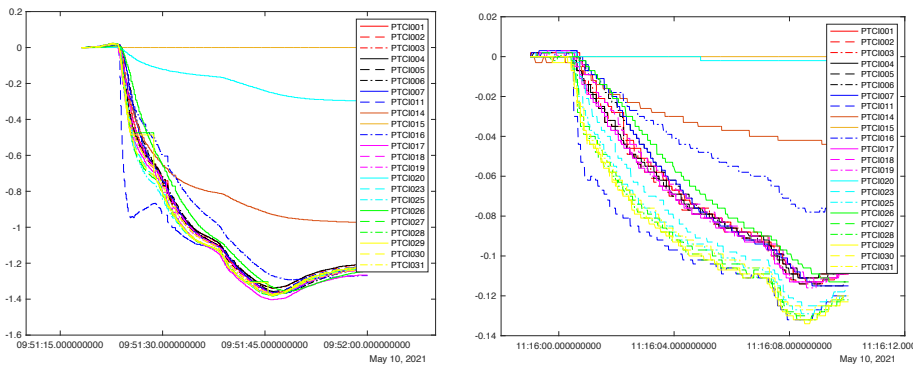


Imagen 5-30: Caídas de presión en rotura de tubería

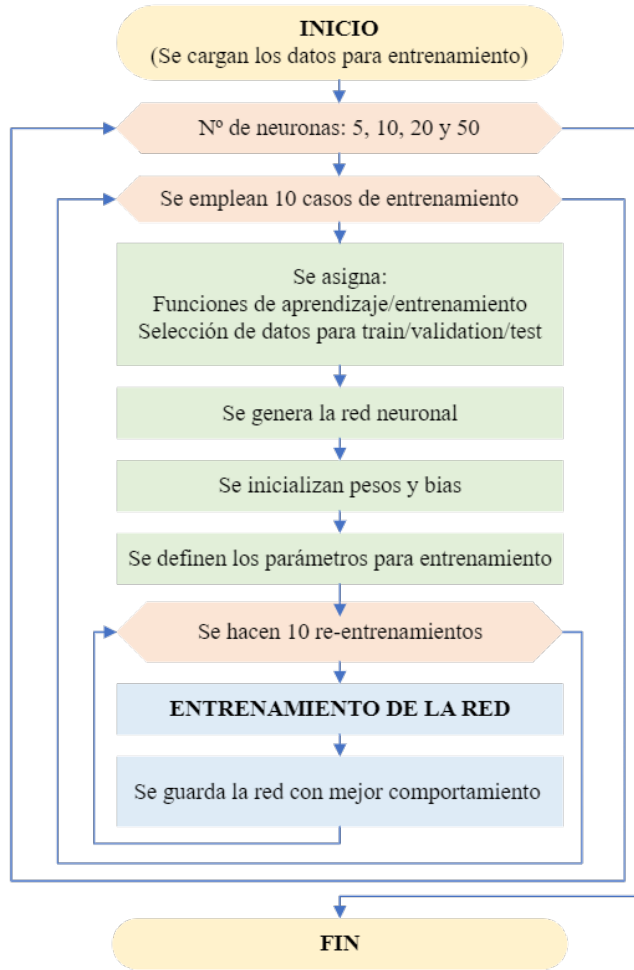
Lo que se hizo, fue trasladar esta información a tablas que puedan ser empleadas como input de la red neuronal. Además, se realizó el etiquetado de los datos, siguiendo el esquema planteado en la imagen 5-28 para localización de los eventos. Se muestran como ejemplo, en la tabla 5-16, los resultados obtenidos para las aperturas realizadas en 2 boquiles contra incendios situados en la misma zona. Como se puede observar, las caídas de presión en los sensores siguen un patrón similar (se ha añadido un código de colores a la tabla, para mejor visualización de los patrones). La red neuronal empleada, tratará de relacionar estos patrones de caída de presión con las diferentes zonas etiquetadas en el buque.

**Tabla 5-8: Caídas de presión relativas (en bares) en los boquiles CI N° 58 y N° 59 (a los 5 s de la apertura y a los 10 segundos de la apertura)**

Sensor	Estación CI N°59		Estación CI N°58	
	Caídas presión tiempo = 5 s	Caídas presión tiempo = 10 s	Caídas presión tiempo = 5 s	Caídas presión tiempo = 10 s
PT-CI001	-0.090	-0.155	-0.079	-0.156
PT-CI002	-0.095	-0.155	-0.086	-0.158
PT-CI003	-0.093	-0.153	-0.086	-0.156
PT-CI004	-0.090	-0.150	-0.090	-0.155
PT-CI005	-0.095	-0.157	-0.088	-0.160
PT-CI006	-0.081	-0.158	-0.077	-0.160
PT-CI007	-0.083	-0.162	-0.077	-0.162
PT-CI008	0	0	0	0
PT-CI010	0	0	0	0
PT-CI011	-0.269	-0.204	-0.239	-0.206
PT-CI012	0	0	0	0
PT-CI013	0	0	0	0
PT-CI014	-0.123	-0.192	-0.108	-0.178
PT-CI015	0	0	0	0
PT-CI016	-0.060	-0.115	-0.048	-0.104
PT-CI017	-0.095	-0.156	-0.094	-0.159
PT-CI018	-0.086	-0.153	-0.086	-0.158
PT-CI019	-0.095	-0.158	-0.090	-0.157
PT-CI020	-0.007	-0.014	-0.007	-0.012
PT-CI023	-0.118	-0.172	-0.111	-0.176
PT-CI025	-0.137	-0.176	-0.129	-0.18
PT-CI026	-0.074	-0.157	-0.065	-0.156
PT-CI027	-0.132	-0.178	-0.127	-0.183
PT-CI028	-0.134	-0.181	-0.127	-0.183
PT-CI029	-0.135	-0.179	-0.130	-0.183
PT-CI030	-0.135	-0.176	-0.127	-0.181
PT-CI031	-0.118	-0.171	-0.118	-0.176

Con estos datos (los datos en este caso indican los retrasos en las señales de los sensores de presión) etiquetados se llevó a cabo un entrenamiento supervisado de una red backpropagation. Se empleó una red para clasificación con el objetivo de validar el uso de esta tipología de redes para la clasificación de las aperturas o roturas del sistema en las zonas del buque etiquetadas. En este caso, la red desarrollada fue algo más sencilla que la empleada en el caso

de la detección de fallos. Se describe la definición y el entrenamiento de la red en la imagen 5-31, empleando un diagrama de flujo:



**Imagen 5-31: Diagrama de flujo del código desarrollado para entrenamiento de la red para localización de fallos**

Se emplearon diferentes funciones de entrenamiento, Scaled Conjugated Gradient, Algoritmo de Levenberg-Marquard y Regularización Bayesiana. Además de ello, en este caso, se emplearon redes de solo 1 capa y se realizaron 10 inicializaciones de cada una de las configuraciones consideradas. Para cada inicialización de la red se realizaron 10 re-entrenos, guardando la capa con mejor comportamiento (performance). En las tablas 5-17, 5-18 y 5-19 se resumen los resultados obtenidos para las configuraciones planteadas.

Tabla 5-9: Resumen de resultados obtenidos para el entrenamiento de la red neuronal empleando la función de entrenamiento: Scaled conjugate gradient

	Neuronas	Matriz de confusión	Curvas ROC (Característica operativa del receptor)																																																																																																																																																
Scaled conjugate gradient	5	<p style="text-align: center;"><b>Confusion Matrix</b></p> <table border="1"> <tr><td>1</td><td>42</td><td>4</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>89.4%</td></tr> <tr><td></td><td>11.5%</td><td>1.1%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.3%</td><td>10.6%</td></tr> <tr><td>2</td><td>0</td><td>11</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>3.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>3</td><td>0</td><td>0</td><td>10</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>2.7%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>4</td><td>2</td><td>0</td><td>0</td><td>83</td><td>0</td><td>3</td><td>0</td><td>88.3%</td></tr> <tr><td></td><td>0.5%</td><td>0.0%</td><td>0.0%</td><td>22.7%</td><td>0.0%</td><td>0.8%</td><td>1.6%</td><td>11.7%</td></tr> <tr><td>5</td><td>0</td><td>0</td><td>4</td><td>0</td><td>45</td><td>1</td><td>0</td><td>50.0%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>1.1%</td><td>0.0%</td><td>12.3%</td><td>0.3%</td><td>0.0%</td><td>10.0%</td></tr> <tr><td>6</td><td>1</td><td>1</td><td>0</td><td>3</td><td>2</td><td>63</td><td>7</td><td>81.8%</td></tr> <tr><td></td><td>0.3%</td><td>0.3%</td><td>0.0%</td><td>0.8%</td><td>0.5%</td><td>17.3%</td><td>1.9%</td><td>18.2%</td></tr> <tr><td>7</td><td>7</td><td>2</td><td>0</td><td>2</td><td>0</td><td>15</td><td>50</td><td>85.8%</td></tr> <tr><td></td><td>1.8%</td><td>0.5%</td><td>0.0%</td><td>0.5%</td><td>0.0%</td><td>4.1%</td><td>13.7%</td><td>34.2%</td></tr> <tr><td></td><td>80.8%</td><td>61.1%</td><td>71.4%</td><td>94.3%</td><td>95.7%</td><td>78.5%</td><td>78.1%</td><td>83.3%</td></tr> <tr><td></td><td>19.2%</td><td>38.9%</td><td>28.6%</td><td>5.7%</td><td>4.3%</td><td>23.2%</td><td>21.9%</td><td>16.7%</td></tr> </table>	1	42	4	0	0	0	0	1	89.4%		11.5%	1.1%	0.0%	0.0%	0.0%	0.0%	0.3%	10.6%	2	0	11	0	0	0	0	0	100%		0.0%	3.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	3	0	0	10	0	0	0	0	100%		0.0%	0.0%	2.7%	0.0%	0.0%	0.0%	0.0%	0.0%	4	2	0	0	83	0	3	0	88.3%		0.5%	0.0%	0.0%	22.7%	0.0%	0.8%	1.6%	11.7%	5	0	0	4	0	45	1	0	50.0%		0.0%	0.0%	1.1%	0.0%	12.3%	0.3%	0.0%	10.0%	6	1	1	0	3	2	63	7	81.8%		0.3%	0.3%	0.0%	0.8%	0.5%	17.3%	1.9%	18.2%	7	7	2	0	2	0	15	50	85.8%		1.8%	0.5%	0.0%	0.5%	0.0%	4.1%	13.7%	34.2%		80.8%	61.1%	71.4%	94.3%	95.7%	78.5%	78.1%	83.3%		19.2%	38.9%	28.6%	5.7%	4.3%	23.2%	21.9%	16.7%	
	1	42	4	0	0	0	0	1	89.4%																																																																																																																																										
		11.5%	1.1%	0.0%	0.0%	0.0%	0.0%	0.3%	10.6%																																																																																																																																										
	2	0	11	0	0	0	0	0	100%																																																																																																																																										
	0.0%	3.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																											
3	0	0	10	0	0	0	0	100%																																																																																																																																											
	0.0%	0.0%	2.7%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																											
4	2	0	0	83	0	3	0	88.3%																																																																																																																																											
	0.5%	0.0%	0.0%	22.7%	0.0%	0.8%	1.6%	11.7%																																																																																																																																											
5	0	0	4	0	45	1	0	50.0%																																																																																																																																											
	0.0%	0.0%	1.1%	0.0%	12.3%	0.3%	0.0%	10.0%																																																																																																																																											
6	1	1	0	3	2	63	7	81.8%																																																																																																																																											
	0.3%	0.3%	0.0%	0.8%	0.5%	17.3%	1.9%	18.2%																																																																																																																																											
7	7	2	0	2	0	15	50	85.8%																																																																																																																																											
	1.8%	0.5%	0.0%	0.5%	0.0%	4.1%	13.7%	34.2%																																																																																																																																											
	80.8%	61.1%	71.4%	94.3%	95.7%	78.5%	78.1%	83.3%																																																																																																																																											
	19.2%	38.9%	28.6%	5.7%	4.3%	23.2%	21.9%	16.7%																																																																																																																																											
10	<p style="text-align: center;"><b>Confusion Matrix</b></p> <table border="1"> <tr><td>1</td><td>42</td><td>3</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>91.3%</td></tr> <tr><td></td><td>11.5%</td><td>0.8%</td><td>0.0%</td><td>0.3%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>8.7%</td></tr> <tr><td>2</td><td>0</td><td>12</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>3.3%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>3</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.3%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>4</td><td>2</td><td>0</td><td>10</td><td>78</td><td>0</td><td>2</td><td>7</td><td>78.8%</td></tr> <tr><td></td><td>0.5%</td><td>0.0%</td><td>2.7%</td><td>21.4%</td><td>0.0%</td><td>0.5%</td><td>1.6%</td><td>21.2%</td></tr> <tr><td>5</td><td>0</td><td>0</td><td>1</td><td>0</td><td>48</td><td>1</td><td>0</td><td>91.8%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.3%</td><td>0.0%</td><td>12.3%</td><td>0.3%</td><td>0.0%</td><td>8.2%</td></tr> <tr><td>6</td><td>3</td><td>1</td><td>0</td><td>8</td><td>2</td><td>62</td><td>7</td><td>74.7%</td></tr> <tr><td></td><td>0.8%</td><td>0.3%</td><td>0.0%</td><td>2.2%</td><td>0.5%</td><td>17.0%</td><td>1.9%</td><td>25.3%</td></tr> <tr><td>7</td><td>5</td><td>2</td><td>0</td><td>1</td><td>0</td><td>17</td><td>50</td><td>56.7%</td></tr> <tr><td></td><td>1.4%</td><td>0.5%</td><td>0.0%</td><td>0.3%</td><td>0.0%</td><td>4.7%</td><td>13.7%</td><td>33.3%</td></tr> <tr><td></td><td>80.8%</td><td>65.7%</td><td>7.1%</td><td>88.6%</td><td>95.7%</td><td>75.6%</td><td>78.1%</td><td>79.9%</td></tr> <tr><td></td><td>19.2%</td><td>33.3%</td><td>92.9%</td><td>11.4%</td><td>4.3%</td><td>24.4%</td><td>21.9%</td><td>20.8%</td></tr> </table>	1	42	3	0	1	0	0	0	91.3%		11.5%	0.8%	0.0%	0.3%	0.0%	0.0%	0.0%	8.7%	2	0	12	0	0	0	0	0	100%		0.0%	3.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	3	0	0	1	0	0	0	0	100%		0.0%	0.0%	0.3%	0.0%	0.0%	0.0%	0.0%	0.0%	4	2	0	10	78	0	2	7	78.8%		0.5%	0.0%	2.7%	21.4%	0.0%	0.5%	1.6%	21.2%	5	0	0	1	0	48	1	0	91.8%		0.0%	0.0%	0.3%	0.0%	12.3%	0.3%	0.0%	8.2%	6	3	1	0	8	2	62	7	74.7%		0.8%	0.3%	0.0%	2.2%	0.5%	17.0%	1.9%	25.3%	7	5	2	0	1	0	17	50	56.7%		1.4%	0.5%	0.0%	0.3%	0.0%	4.7%	13.7%	33.3%		80.8%	65.7%	7.1%	88.6%	95.7%	75.6%	78.1%	79.9%		19.2%	33.3%	92.9%	11.4%	4.3%	24.4%	21.9%	20.8%		
1	42	3	0	1	0	0	0	91.3%																																																																																																																																											
	11.5%	0.8%	0.0%	0.3%	0.0%	0.0%	0.0%	8.7%																																																																																																																																											
2	0	12	0	0	0	0	0	100%																																																																																																																																											
	0.0%	3.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																											
3	0	0	1	0	0	0	0	100%																																																																																																																																											
	0.0%	0.0%	0.3%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																											
4	2	0	10	78	0	2	7	78.8%																																																																																																																																											
	0.5%	0.0%	2.7%	21.4%	0.0%	0.5%	1.6%	21.2%																																																																																																																																											
5	0	0	1	0	48	1	0	91.8%																																																																																																																																											
	0.0%	0.0%	0.3%	0.0%	12.3%	0.3%	0.0%	8.2%																																																																																																																																											
6	3	1	0	8	2	62	7	74.7%																																																																																																																																											
	0.8%	0.3%	0.0%	2.2%	0.5%	17.0%	1.9%	25.3%																																																																																																																																											
7	5	2	0	1	0	17	50	56.7%																																																																																																																																											
	1.4%	0.5%	0.0%	0.3%	0.0%	4.7%	13.7%	33.3%																																																																																																																																											
	80.8%	65.7%	7.1%	88.6%	95.7%	75.6%	78.1%	79.9%																																																																																																																																											
	19.2%	33.3%	92.9%	11.4%	4.3%	24.4%	21.9%	20.8%																																																																																																																																											
20	<p style="text-align: center;"><b>Confusion Matrix</b></p> <table border="1"> <tr><td>1</td><td>43</td><td>3</td><td>0</td><td>0</td><td>0</td><td>2</td><td>0</td><td>89.0%</td></tr> <tr><td></td><td>11.8%</td><td>0.8%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.5%</td><td>0.0%</td><td>10.4%</td></tr> <tr><td>2</td><td>0</td><td>12</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>3.3%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>3</td><td>0</td><td>0</td><td>6</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>1.6%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>4</td><td>1</td><td>0</td><td>0</td><td>74</td><td>0</td><td>3</td><td>6</td><td>88.1%</td></tr> <tr><td></td><td>0.3%</td><td>0.0%</td><td>0.0%</td><td>20.3%</td><td>0.0%</td><td>0.8%</td><td>1.6%</td><td>11.9%</td></tr> <tr><td>5</td><td>0</td><td>0</td><td>4</td><td>0</td><td>44</td><td>1</td><td>0</td><td>89.8%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>1.1%</td><td>0.0%</td><td>12.1%</td><td>0.3%</td><td>0.0%</td><td>10.2%</td></tr> <tr><td>6</td><td>3</td><td>1</td><td>4</td><td>5</td><td>3</td><td>60</td><td>8</td><td>71.4%</td></tr> <tr><td></td><td>0.8%</td><td>0.3%</td><td>1.1%</td><td>1.4%</td><td>0.8%</td><td>16.4%</td><td>2.2%</td><td>26.6%</td></tr> <tr><td>7</td><td>5</td><td>2</td><td>0</td><td>0</td><td>0</td><td>16</td><td>50</td><td>81.0%</td></tr> <tr><td></td><td>1.4%</td><td>0.5%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>4.4%</td><td>13.7%</td><td>20.0%</td></tr> <tr><td></td><td>82.7%</td><td>65.7%</td><td>42.0%</td><td>84.1%</td><td>93.0%</td><td>73.2%</td><td>78.1%</td><td>78.2%</td></tr> <tr><td></td><td>17.3%</td><td>33.3%</td><td>57.1%</td><td>15.9%</td><td>6.4%</td><td>26.8%</td><td>21.9%</td><td>20.8%</td></tr> </table>	1	43	3	0	0	0	2	0	89.0%		11.8%	0.8%	0.0%	0.0%	0.0%	0.5%	0.0%	10.4%	2	0	12	0	0	0	0	0	100%		0.0%	3.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	3	0	0	6	0	0	0	0	100%		0.0%	0.0%	1.6%	0.0%	0.0%	0.0%	0.0%	0.0%	4	1	0	0	74	0	3	6	88.1%		0.3%	0.0%	0.0%	20.3%	0.0%	0.8%	1.6%	11.9%	5	0	0	4	0	44	1	0	89.8%		0.0%	0.0%	1.1%	0.0%	12.1%	0.3%	0.0%	10.2%	6	3	1	4	5	3	60	8	71.4%		0.8%	0.3%	1.1%	1.4%	0.8%	16.4%	2.2%	26.6%	7	5	2	0	0	0	16	50	81.0%		1.4%	0.5%	0.0%	0.0%	0.0%	4.4%	13.7%	20.0%		82.7%	65.7%	42.0%	84.1%	93.0%	73.2%	78.1%	78.2%		17.3%	33.3%	57.1%	15.9%	6.4%	26.8%	21.9%	20.8%		
1	43	3	0	0	0	2	0	89.0%																																																																																																																																											
	11.8%	0.8%	0.0%	0.0%	0.0%	0.5%	0.0%	10.4%																																																																																																																																											
2	0	12	0	0	0	0	0	100%																																																																																																																																											
	0.0%	3.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																											
3	0	0	6	0	0	0	0	100%																																																																																																																																											
	0.0%	0.0%	1.6%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																											
4	1	0	0	74	0	3	6	88.1%																																																																																																																																											
	0.3%	0.0%	0.0%	20.3%	0.0%	0.8%	1.6%	11.9%																																																																																																																																											
5	0	0	4	0	44	1	0	89.8%																																																																																																																																											
	0.0%	0.0%	1.1%	0.0%	12.1%	0.3%	0.0%	10.2%																																																																																																																																											
6	3	1	4	5	3	60	8	71.4%																																																																																																																																											
	0.8%	0.3%	1.1%	1.4%	0.8%	16.4%	2.2%	26.6%																																																																																																																																											
7	5	2	0	0	0	16	50	81.0%																																																																																																																																											
	1.4%	0.5%	0.0%	0.0%	0.0%	4.4%	13.7%	20.0%																																																																																																																																											
	82.7%	65.7%	42.0%	84.1%	93.0%	73.2%	78.1%	78.2%																																																																																																																																											
	17.3%	33.3%	57.1%	15.9%	6.4%	26.8%	21.9%	20.8%																																																																																																																																											
50	<p style="text-align: center;"><b>Confusion Matrix</b></p> <table border="1"> <tr><td>1</td><td>40</td><td>2</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>95.2%</td></tr> <tr><td></td><td>11.0%</td><td>0.5%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>4.8%</td></tr> <tr><td>2</td><td>0</td><td>12</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>3.3%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>3</td><td>0</td><td>0</td><td>6</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>1.6%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>4</td><td>3</td><td>1</td><td>4</td><td>75</td><td>0</td><td>4</td><td>12</td><td>75.8%</td></tr> <tr><td></td><td>0.8%</td><td>0.3%</td><td>1.1%</td><td>20.5%</td><td>0.0%</td><td>1.1%</td><td>3.3%</td><td>24.2%</td></tr> <tr><td>5</td><td>0</td><td>0</td><td>4</td><td>0</td><td>44</td><td>1</td><td>0</td><td>89.8%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>1.1%</td><td>0.0%</td><td>12.1%</td><td>0.3%</td><td>0.0%</td><td>10.2%</td></tr> <tr><td>6</td><td>5</td><td>1</td><td>0</td><td>11</td><td>3</td><td>65</td><td>8</td><td>69.9%</td></tr> <tr><td></td><td>1.4%</td><td>0.3%</td><td>0.0%</td><td>3.0%</td><td>0.8%</td><td>17.5%</td><td>2.2%</td><td>30.1%</td></tr> <tr><td>7</td><td>4</td><td>2</td><td>0</td><td>2</td><td>0</td><td>12</td><td>44</td><td>68.8%</td></tr> <tr><td></td><td>1.1%</td><td>0.5%</td><td>0.0%</td><td>0.5%</td><td>0.0%</td><td>3.3%</td><td>12.1%</td><td>31.3%</td></tr> <tr><td></td><td>76.9%</td><td>66.7%</td><td>42.0%</td><td>85.2%</td><td>93.0%</td><td>73.3%</td><td>68.8%</td><td>78.4%</td></tr> <tr><td></td><td>23.1%</td><td>33.3%</td><td>57.1%</td><td>14.8%</td><td>6.4%</td><td>26.7%</td><td>31.3%</td><td>21.6%</td></tr> </table>	1	40	2	0	0	0	0	0	95.2%		11.0%	0.5%	0.0%	0.0%	0.0%	0.0%	0.0%	4.8%	2	0	12	0	0	0	0	0	100%		0.0%	3.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	3	0	0	6	0	0	0	0	100%		0.0%	0.0%	1.6%	0.0%	0.0%	0.0%	0.0%	0.0%	4	3	1	4	75	0	4	12	75.8%		0.8%	0.3%	1.1%	20.5%	0.0%	1.1%	3.3%	24.2%	5	0	0	4	0	44	1	0	89.8%		0.0%	0.0%	1.1%	0.0%	12.1%	0.3%	0.0%	10.2%	6	5	1	0	11	3	65	8	69.9%		1.4%	0.3%	0.0%	3.0%	0.8%	17.5%	2.2%	30.1%	7	4	2	0	2	0	12	44	68.8%		1.1%	0.5%	0.0%	0.5%	0.0%	3.3%	12.1%	31.3%		76.9%	66.7%	42.0%	85.2%	93.0%	73.3%	68.8%	78.4%		23.1%	33.3%	57.1%	14.8%	6.4%	26.7%	31.3%	21.6%		
1	40	2	0	0	0	0	0	95.2%																																																																																																																																											
	11.0%	0.5%	0.0%	0.0%	0.0%	0.0%	0.0%	4.8%																																																																																																																																											
2	0	12	0	0	0	0	0	100%																																																																																																																																											
	0.0%	3.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																											
3	0	0	6	0	0	0	0	100%																																																																																																																																											
	0.0%	0.0%	1.6%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																											
4	3	1	4	75	0	4	12	75.8%																																																																																																																																											
	0.8%	0.3%	1.1%	20.5%	0.0%	1.1%	3.3%	24.2%																																																																																																																																											
5	0	0	4	0	44	1	0	89.8%																																																																																																																																											
	0.0%	0.0%	1.1%	0.0%	12.1%	0.3%	0.0%	10.2%																																																																																																																																											
6	5	1	0	11	3	65	8	69.9%																																																																																																																																											
	1.4%	0.3%	0.0%	3.0%	0.8%	17.5%	2.2%	30.1%																																																																																																																																											
7	4	2	0	2	0	12	44	68.8%																																																																																																																																											
	1.1%	0.5%	0.0%	0.5%	0.0%	3.3%	12.1%	31.3%																																																																																																																																											
	76.9%	66.7%	42.0%	85.2%	93.0%	73.3%	68.8%	78.4%																																																																																																																																											
	23.1%	33.3%	57.1%	14.8%	6.4%	26.7%	31.3%	21.6%																																																																																																																																											



Tabla 5-10: Resumen de resultados obtenidos para el entrenamiento de la red neuronal empleando la función de entrenamiento: Levenberg-Marquard

	Neuronas	Matriz de confusión	Curvas ROC (Característica operativa del receptor)																																																																																																																																																	
Levenberg-Marquard	5	<p style="text-align: center;"><b>Confusion Matrix</b></p> <table border="1"> <tr><td>1</td><td>50</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>13.7%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>2</td><td>0</td><td>18</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>4.5%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>3</td><td>2</td><td>0</td><td>14</td><td>1</td><td>0</td><td>0</td><td>0</td><td>82.4%</td></tr> <tr><td></td><td>0.5%</td><td>0.0%</td><td>3.5%</td><td>0.3%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>17.6%</td></tr> <tr><td>4</td><td>0</td><td>0</td><td>0</td><td>86</td><td>0</td><td>0</td><td>1</td><td>88.9%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>23.6%</td><td>0.0%</td><td>0.0%</td><td>0.3%</td><td>1.1%</td></tr> <tr><td>5</td><td>0</td><td>0</td><td>0</td><td>0</td><td>47</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>12.9%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>6</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>82</td><td>3</td><td>95.3%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.3%</td><td>0.0%</td><td>22.5%</td><td>0.8%</td><td>4.7%</td></tr> <tr><td>7</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>60</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>16.4%</td><td>0.0%</td></tr> <tr><td></td><td>98.2%</td><td>100%</td><td>100%</td><td>97.7%</td><td>100%</td><td>100%</td><td>93.8%</td><td>97.8%</td></tr> <tr><td></td><td>3.8%</td><td>0.0%</td><td>0.0%</td><td>2.3%</td><td>0.0%</td><td>0.0%</td><td>6.3%</td><td>2.2%</td></tr> </table>	1	50	0	0	0	0	0	0	100%		13.7%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	2	0	18	0	0	0	0	0	100%		0.0%	4.5%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	3	2	0	14	1	0	0	0	82.4%		0.5%	0.0%	3.5%	0.3%	0.0%	0.0%	0.0%	17.6%	4	0	0	0	86	0	0	1	88.9%		0.0%	0.0%	0.0%	23.6%	0.0%	0.0%	0.3%	1.1%	5	0	0	0	0	47	0	0	100%		0.0%	0.0%	0.0%	0.0%	12.9%	0.0%	0.0%	0.0%	6	0	0	0	1	0	82	3	95.3%		0.0%	0.0%	0.0%	0.3%	0.0%	22.5%	0.8%	4.7%	7	0	0	0	0	0	0	60	100%		0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	16.4%	0.0%		98.2%	100%	100%	97.7%	100%	100%	93.8%	97.8%		3.8%	0.0%	0.0%	2.3%	0.0%	0.0%	6.3%	2.2%		
	1	50	0	0	0	0	0	0	100%																																																																																																																																											
		13.7%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																											
	2	0	18	0	0	0	0	0	100%																																																																																																																																											
		0.0%	4.5%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																											
3	2	0	14	1	0	0	0	82.4%																																																																																																																																												
	0.5%	0.0%	3.5%	0.3%	0.0%	0.0%	0.0%	17.6%																																																																																																																																												
4	0	0	0	86	0	0	1	88.9%																																																																																																																																												
	0.0%	0.0%	0.0%	23.6%	0.0%	0.0%	0.3%	1.1%																																																																																																																																												
5	0	0	0	0	47	0	0	100%																																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	12.9%	0.0%	0.0%	0.0%																																																																																																																																												
6	0	0	0	1	0	82	3	95.3%																																																																																																																																												
	0.0%	0.0%	0.0%	0.3%	0.0%	22.5%	0.8%	4.7%																																																																																																																																												
7	0	0	0	0	0	0	60	100%																																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	16.4%	0.0%																																																																																																																																												
	98.2%	100%	100%	97.7%	100%	100%	93.8%	97.8%																																																																																																																																												
	3.8%	0.0%	0.0%	2.3%	0.0%	0.0%	6.3%	2.2%																																																																																																																																												
10	<p style="text-align: center;"><b>Confusion Matrix</b></p> <table border="1"> <tr><td>1</td><td>51</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>98.1%</td></tr> <tr><td></td><td>14.0%</td><td>0.3%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>1.0%</td></tr> <tr><td>2</td><td>0</td><td>16</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>4.4%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>3</td><td>0</td><td>0</td><td>14</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>3.6%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>4</td><td>0</td><td>0</td><td>0</td><td>88</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>24.1%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>5</td><td>0</td><td>0</td><td>0</td><td>0</td><td>46</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>12.6%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>6</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>82</td><td>1</td><td>96.0%</td></tr> <tr><td></td><td>0.3%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.3%</td><td>22.5%</td><td>0.3%</td><td>3.5%</td></tr> <tr><td>7</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>63</td><td>98.4%</td></tr> <tr><td></td><td>0.0%</td><td>0.3%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>17.3%</td><td>1.9%</td></tr> <tr><td></td><td>88.1%</td><td>88.9%</td><td>100%</td><td>100%</td><td>97.5%</td><td>100%</td><td>98.4%</td><td>98.6%</td></tr> <tr><td></td><td>1.9%</td><td>11.1%</td><td>0.0%</td><td>0.0%</td><td>2.5%</td><td>0.0%</td><td>1.6%</td><td>5.6%</td></tr> </table>	1	51	0	0	0	0	0	0	98.1%		14.0%	0.3%	0.0%	0.0%	0.0%	0.0%	0.0%	1.0%	2	0	16	0	0	0	0	0	100%		0.0%	4.4%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	3	0	0	14	0	0	0	0	100%		0.0%	0.0%	3.6%	0.0%	0.0%	0.0%	0.0%	0.0%	4	0	0	0	88	0	0	0	100%		0.0%	0.0%	0.0%	24.1%	0.0%	0.0%	0.0%	0.0%	5	0	0	0	0	46	0	0	100%		0.0%	0.0%	0.0%	0.0%	12.6%	0.0%	0.0%	0.0%	6	1	0	0	0	1	82	1	96.0%		0.3%	0.0%	0.0%	0.0%	0.3%	22.5%	0.3%	3.5%	7	0	1	0	0	0	0	63	98.4%		0.0%	0.3%	0.0%	0.0%	0.0%	0.0%	17.3%	1.9%		88.1%	88.9%	100%	100%	97.5%	100%	98.4%	98.6%		1.9%	11.1%	0.0%	0.0%	2.5%	0.0%	1.6%	5.6%			
1	51	0	0	0	0	0	0	98.1%																																																																																																																																												
	14.0%	0.3%	0.0%	0.0%	0.0%	0.0%	0.0%	1.0%																																																																																																																																												
2	0	16	0	0	0	0	0	100%																																																																																																																																												
	0.0%	4.4%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																												
3	0	0	14	0	0	0	0	100%																																																																																																																																												
	0.0%	0.0%	3.6%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																												
4	0	0	0	88	0	0	0	100%																																																																																																																																												
	0.0%	0.0%	0.0%	24.1%	0.0%	0.0%	0.0%	0.0%																																																																																																																																												
5	0	0	0	0	46	0	0	100%																																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	12.6%	0.0%	0.0%	0.0%																																																																																																																																												
6	1	0	0	0	1	82	1	96.0%																																																																																																																																												
	0.3%	0.0%	0.0%	0.0%	0.3%	22.5%	0.3%	3.5%																																																																																																																																												
7	0	1	0	0	0	0	63	98.4%																																																																																																																																												
	0.0%	0.3%	0.0%	0.0%	0.0%	0.0%	17.3%	1.9%																																																																																																																																												
	88.1%	88.9%	100%	100%	97.5%	100%	98.4%	98.6%																																																																																																																																												
	1.9%	11.1%	0.0%	0.0%	2.5%	0.0%	1.6%	5.6%																																																																																																																																												
20	<p style="text-align: center;"><b>Confusion Matrix</b></p> <table border="1"> <tr><td>1</td><td>49</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>13.4%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>2</td><td>0</td><td>17</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>4.7%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>3</td><td>0</td><td>0</td><td>14</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>3.6%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>4</td><td>0</td><td>0</td><td>0</td><td>88</td><td>0</td><td>0</td><td>1</td><td>97.8%</td></tr> <tr><td></td><td>0.0%</td><td>0.3%</td><td>0.0%</td><td>24.1%</td><td>0.0%</td><td>0.0%</td><td>0.3%</td><td>2.2%</td></tr> <tr><td>5</td><td>0</td><td>0</td><td>0</td><td>0</td><td>47</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>12.9%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>6</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>81</td><td>0</td><td>98.0%</td></tr> <tr><td></td><td>0.3%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>22.2%</td><td>0.0%</td><td>1.2%</td></tr> <tr><td>7</td><td>2</td><td>0</td><td>0</td><td>0</td><td>1</td><td>63</td><td>0</td><td>95.0%</td></tr> <tr><td></td><td>0.5%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.3%</td><td>17.3%</td><td>0.0%</td><td>4.7%</td></tr> <tr><td></td><td>94.2%</td><td>94.4%</td><td>100%</td><td>100%</td><td>100%</td><td>98.8%</td><td>98.4%</td><td>88.4%</td></tr> <tr><td></td><td>5.8%</td><td>5.6%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>1.2%</td><td>1.6%</td><td>1.6%</td></tr> </table>	1	49	0	0	0	0	0	0	100%		13.4%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	2	0	17	0	0	0	0	0	100%		0.0%	4.7%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	3	0	0	14	0	0	0	0	100%		0.0%	0.0%	3.6%	0.0%	0.0%	0.0%	0.0%	0.0%	4	0	0	0	88	0	0	1	97.8%		0.0%	0.3%	0.0%	24.1%	0.0%	0.0%	0.3%	2.2%	5	0	0	0	0	47	0	0	100%		0.0%	0.0%	0.0%	0.0%	12.9%	0.0%	0.0%	0.0%	6	1	0	0	0	0	81	0	98.0%		0.3%	0.0%	0.0%	0.0%	0.0%	22.2%	0.0%	1.2%	7	2	0	0	0	1	63	0	95.0%		0.5%	0.0%	0.0%	0.0%	0.3%	17.3%	0.0%	4.7%		94.2%	94.4%	100%	100%	100%	98.8%	98.4%	88.4%		5.8%	5.6%	0.0%	0.0%	0.0%	1.2%	1.6%	1.6%			
1	49	0	0	0	0	0	0	100%																																																																																																																																												
	13.4%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																												
2	0	17	0	0	0	0	0	100%																																																																																																																																												
	0.0%	4.7%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																												
3	0	0	14	0	0	0	0	100%																																																																																																																																												
	0.0%	0.0%	3.6%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																												
4	0	0	0	88	0	0	1	97.8%																																																																																																																																												
	0.0%	0.3%	0.0%	24.1%	0.0%	0.0%	0.3%	2.2%																																																																																																																																												
5	0	0	0	0	47	0	0	100%																																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	12.9%	0.0%	0.0%	0.0%																																																																																																																																												
6	1	0	0	0	0	81	0	98.0%																																																																																																																																												
	0.3%	0.0%	0.0%	0.0%	0.0%	22.2%	0.0%	1.2%																																																																																																																																												
7	2	0	0	0	1	63	0	95.0%																																																																																																																																												
	0.5%	0.0%	0.0%	0.0%	0.3%	17.3%	0.0%	4.7%																																																																																																																																												
	94.2%	94.4%	100%	100%	100%	98.8%	98.4%	88.4%																																																																																																																																												
	5.8%	5.6%	0.0%	0.0%	0.0%	1.2%	1.6%	1.6%																																																																																																																																												
50	<p style="text-align: center;"><b>Confusion Matrix</b></p> <table border="1"> <tr><td>1</td><td>49</td><td>2</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>96.1%</td></tr> <tr><td></td><td>13.4%</td><td>0.5%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>3.9%</td></tr> <tr><td>2</td><td>0</td><td>14</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>3.6%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>3</td><td>0</td><td>0</td><td>14</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>3.6%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>4</td><td>0</td><td>0</td><td>0</td><td>86</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>23.6%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>5</td><td>0</td><td>0</td><td>0</td><td>0</td><td>47</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>12.9%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>6</td><td>1</td><td>0</td><td>0</td><td>0</td><td>2</td><td>0</td><td>82</td><td>1</td><td>95.3%</td></tr> <tr><td></td><td>0.3%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.5%</td><td>0.0%</td><td>22.5%</td><td>0.3%</td><td>4.7%</td></tr> <tr><td>7</td><td>2</td><td>2</td><td>0</td><td>0</td><td>0</td><td>0</td><td>63</td><td>91.0%</td></tr> <tr><td></td><td>0.5%</td><td>0.5%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>17.3%</td><td>6.0%</td></tr> <tr><td></td><td>94.2%</td><td>77.0%</td><td>100%</td><td>97.7%</td><td>100%</td><td>100%</td><td>98.4%</td><td>97.3%</td></tr> <tr><td></td><td>5.8%</td><td>22.2%</td><td>0.0%</td><td>2.3%</td><td>0.0%</td><td>0.0%</td><td>1.6%</td><td>2.7%</td></tr> </table>	1	49	2	0	0	0	0	0	96.1%		13.4%	0.5%	0.0%	0.0%	0.0%	0.0%	0.0%	3.9%	2	0	14	0	0	0	0	0	100%		0.0%	3.6%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	3	0	0	14	0	0	0	0	100%		0.0%	0.0%	3.6%	0.0%	0.0%	0.0%	0.0%	0.0%	4	0	0	0	86	0	0	0	100%		0.0%	0.0%	0.0%	23.6%	0.0%	0.0%	0.0%	0.0%	5	0	0	0	0	47	0	0	100%		0.0%	0.0%	0.0%	0.0%	12.9%	0.0%	0.0%	0.0%	6	1	0	0	0	2	0	82	1	95.3%		0.3%	0.0%	0.0%	0.0%	0.5%	0.0%	22.5%	0.3%	4.7%	7	2	2	0	0	0	0	63	91.0%		0.5%	0.5%	0.0%	0.0%	0.0%	0.0%	17.3%	6.0%		94.2%	77.0%	100%	97.7%	100%	100%	98.4%	97.3%		5.8%	22.2%	0.0%	2.3%	0.0%	0.0%	1.6%	2.7%	
1	49	2	0	0	0	0	0	96.1%																																																																																																																																												
	13.4%	0.5%	0.0%	0.0%	0.0%	0.0%	0.0%	3.9%																																																																																																																																												
2	0	14	0	0	0	0	0	100%																																																																																																																																												
	0.0%	3.6%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																												
3	0	0	14	0	0	0	0	100%																																																																																																																																												
	0.0%	0.0%	3.6%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																																												
4	0	0	0	86	0	0	0	100%																																																																																																																																												
	0.0%	0.0%	0.0%	23.6%	0.0%	0.0%	0.0%	0.0%																																																																																																																																												
5	0	0	0	0	47	0	0	100%																																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	12.9%	0.0%	0.0%	0.0%																																																																																																																																												
6	1	0	0	0	2	0	82	1	95.3%																																																																																																																																											
	0.3%	0.0%	0.0%	0.0%	0.5%	0.0%	22.5%	0.3%	4.7%																																																																																																																																											
7	2	2	0	0	0	0	63	91.0%																																																																																																																																												
	0.5%	0.5%	0.0%	0.0%	0.0%	0.0%	17.3%	6.0%																																																																																																																																												
	94.2%	77.0%	100%	97.7%	100%	100%	98.4%	97.3%																																																																																																																																												
	5.8%	22.2%	0.0%	2.3%	0.0%	0.0%	1.6%	2.7%																																																																																																																																												

Tabla 5-11: Resumen de resultados obtenidos para el entrenamiento de la red neuronal empleando la función de entrenamiento: Regularización Bayesiana

	Neuronas	Matriz de confusión	Curvas ROC (Característica operativa del receptor)																																																																																																																																
Regularización Bayesiana	5	<p style="text-align: center;"><b>Confusion Matrix</b></p> <table border="1"> <tr><td>1</td><td>51</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>14.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>2</td><td>0</td><td>18</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>4.9%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>3</td><td>0</td><td>0</td><td>14</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>3.8%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>4</td><td>0</td><td>0</td><td>0</td><td>88</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>24.1%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>5</td><td>0</td><td>0</td><td>0</td><td>0</td><td>47</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>12.9%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>6</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>82</td><td>100%</td></tr> <tr><td></td><td>0.3%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>22.5%</td><td>0.0%</td></tr> <tr><td>7</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td></td><td>98.1%</td><td>100%</td><td>100%</td><td>100%</td><td>100%</td><td>100%</td><td>99.9%</td></tr> <tr><td></td><td>3.9%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> </table>	1	51	0	0	0	0	0	100%		14.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	2	0	18	0	0	0	0	100%		0.0%	4.9%	0.0%	0.0%	0.0%	0.0%	0.0%	3	0	0	14	0	0	0	100%		0.0%	0.0%	3.8%	0.0%	0.0%	0.0%	0.0%	4	0	0	0	88	0	0	100%		0.0%	0.0%	0.0%	24.1%	0.0%	0.0%	0.0%	5	0	0	0	0	47	0	100%		0.0%	0.0%	0.0%	0.0%	12.9%	0.0%	0.0%	6	0	0	0	0	0	82	100%		0.3%	0.0%	0.0%	0.0%	0.0%	22.5%	0.0%	7	0	0	0	0	0	0	100%		0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%		98.1%	100%	100%	100%	100%	100%	99.9%		3.9%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	
	1	51	0	0	0	0	0	100%																																																																																																																											
		14.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																											
	2	0	18	0	0	0	0	100%																																																																																																																											
	0.0%	4.9%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
3	0	0	14	0	0	0	100%																																																																																																																												
	0.0%	0.0%	3.8%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
4	0	0	0	88	0	0	100%																																																																																																																												
	0.0%	0.0%	0.0%	24.1%	0.0%	0.0%	0.0%																																																																																																																												
5	0	0	0	0	47	0	100%																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	12.9%	0.0%	0.0%																																																																																																																												
6	0	0	0	0	0	82	100%																																																																																																																												
	0.3%	0.0%	0.0%	0.0%	0.0%	22.5%	0.0%																																																																																																																												
7	0	0	0	0	0	0	100%																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
	98.1%	100%	100%	100%	100%	100%	99.9%																																																																																																																												
	3.9%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
10	<p style="text-align: center;"><b>Confusion Matrix</b></p> <table border="1"> <tr><td>1</td><td>52</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>14.2%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>2</td><td>0</td><td>17</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>4.7%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>3</td><td>0</td><td>0</td><td>14</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>3.8%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>4</td><td>0</td><td>0</td><td>0</td><td>88</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>24.1%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>5</td><td>0</td><td>0</td><td>0</td><td>0</td><td>46</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>12.8%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>6</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>82</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.3%</td><td>22.5%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>7</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.3%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td></td><td>100%</td><td>94.4%</td><td>100%</td><td>100%</td><td>97.9%</td><td>100%</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>5.6%</td><td>0.0%</td><td>0.0%</td><td>2.1%</td><td>0.0%</td><td>0.0%</td></tr> </table>	1	52	0	0	0	0	0	100%		14.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	2	0	17	0	0	0	0	100%		0.0%	4.7%	0.0%	0.0%	0.0%	0.0%	0.0%	3	0	0	14	0	0	0	100%		0.0%	0.0%	3.8%	0.0%	0.0%	0.0%	0.0%	4	0	0	0	88	0	0	100%		0.0%	0.0%	0.0%	24.1%	0.0%	0.0%	0.0%	5	0	0	0	0	46	0	100%		0.0%	0.0%	0.0%	0.0%	12.8%	0.0%	0.0%	6	0	0	0	0	0	82	100%		0.0%	0.0%	0.0%	0.3%	22.5%	0.0%	0.0%	7	0	0	0	0	0	0	100%		0.0%	0.3%	0.0%	0.0%	0.0%	0.0%	0.0%		100%	94.4%	100%	100%	97.9%	100%	100%		0.0%	5.6%	0.0%	0.0%	2.1%	0.0%	0.0%		
1	52	0	0	0	0	0	100%																																																																																																																												
	14.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
2	0	17	0	0	0	0	100%																																																																																																																												
	0.0%	4.7%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
3	0	0	14	0	0	0	100%																																																																																																																												
	0.0%	0.0%	3.8%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
4	0	0	0	88	0	0	100%																																																																																																																												
	0.0%	0.0%	0.0%	24.1%	0.0%	0.0%	0.0%																																																																																																																												
5	0	0	0	0	46	0	100%																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	12.8%	0.0%	0.0%																																																																																																																												
6	0	0	0	0	0	82	100%																																																																																																																												
	0.0%	0.0%	0.0%	0.3%	22.5%	0.0%	0.0%																																																																																																																												
7	0	0	0	0	0	0	100%																																																																																																																												
	0.0%	0.3%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
	100%	94.4%	100%	100%	97.9%	100%	100%																																																																																																																												
	0.0%	5.6%	0.0%	0.0%	2.1%	0.0%	0.0%																																																																																																																												
20	<p style="text-align: center;"><b>Confusion Matrix</b></p> <table border="1"> <tr><td>1</td><td>52</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>14.2%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>2</td><td>0</td><td>18</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>4.9%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>3</td><td>0</td><td>0</td><td>14</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>3.8%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>4</td><td>0</td><td>0</td><td>0</td><td>88</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>24.1%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>5</td><td>0</td><td>0</td><td>0</td><td>0</td><td>47</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>12.9%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>6</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>82</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>22.5%</td><td>0.0%</td></tr> <tr><td>7</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td></td><td>100%</td><td>100%</td><td>100%</td><td>100%</td><td>100%</td><td>100%</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> </table>	1	52	0	0	0	0	0	100%		14.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	2	0	18	0	0	0	0	100%		0.0%	4.9%	0.0%	0.0%	0.0%	0.0%	0.0%	3	0	0	14	0	0	0	100%		0.0%	0.0%	3.8%	0.0%	0.0%	0.0%	0.0%	4	0	0	0	88	0	0	100%		0.0%	0.0%	0.0%	24.1%	0.0%	0.0%	0.0%	5	0	0	0	0	47	0	100%		0.0%	0.0%	0.0%	0.0%	12.9%	0.0%	0.0%	6	0	0	0	0	0	82	100%		0.0%	0.0%	0.0%	0.0%	0.0%	22.5%	0.0%	7	0	0	0	0	0	0	100%		0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%		100%	100%	100%	100%	100%	100%	100%		0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%		
1	52	0	0	0	0	0	100%																																																																																																																												
	14.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
2	0	18	0	0	0	0	100%																																																																																																																												
	0.0%	4.9%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
3	0	0	14	0	0	0	100%																																																																																																																												
	0.0%	0.0%	3.8%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
4	0	0	0	88	0	0	100%																																																																																																																												
	0.0%	0.0%	0.0%	24.1%	0.0%	0.0%	0.0%																																																																																																																												
5	0	0	0	0	47	0	100%																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	12.9%	0.0%	0.0%																																																																																																																												
6	0	0	0	0	0	82	100%																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	0.0%	22.5%	0.0%																																																																																																																												
7	0	0	0	0	0	0	100%																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
	100%	100%	100%	100%	100%	100%	100%																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
50	<p style="text-align: center;"><b>Confusion Matrix</b></p> <table border="1"> <tr><td>1</td><td>52</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>14.2%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>2</td><td>0</td><td>18</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>4.9%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>3</td><td>0</td><td>0</td><td>14</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>3.8%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>4</td><td>0</td><td>0</td><td>0</td><td>88</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>24.1%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>5</td><td>0</td><td>0</td><td>0</td><td>0</td><td>47</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>12.9%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td>6</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>82</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>22.5%</td><td>0.0%</td></tr> <tr><td>7</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> <tr><td></td><td>100%</td><td>100%</td><td>100%</td><td>100%</td><td>100%</td><td>100%</td><td>100%</td></tr> <tr><td></td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td><td>0.0%</td></tr> </table>	1	52	0	0	0	0	0	100%		14.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	2	0	18	0	0	0	0	100%		0.0%	4.9%	0.0%	0.0%	0.0%	0.0%	0.0%	3	0	0	14	0	0	0	100%		0.0%	0.0%	3.8%	0.0%	0.0%	0.0%	0.0%	4	0	0	0	88	0	0	100%		0.0%	0.0%	0.0%	24.1%	0.0%	0.0%	0.0%	5	0	0	0	0	47	0	100%		0.0%	0.0%	0.0%	0.0%	12.9%	0.0%	0.0%	6	0	0	0	0	0	82	100%		0.0%	0.0%	0.0%	0.0%	0.0%	22.5%	0.0%	7	0	0	0	0	0	0	100%		0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%		100%	100%	100%	100%	100%	100%	100%		0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%		
1	52	0	0	0	0	0	100%																																																																																																																												
	14.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
2	0	18	0	0	0	0	100%																																																																																																																												
	0.0%	4.9%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
3	0	0	14	0	0	0	100%																																																																																																																												
	0.0%	0.0%	3.8%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
4	0	0	0	88	0	0	100%																																																																																																																												
	0.0%	0.0%	0.0%	24.1%	0.0%	0.0%	0.0%																																																																																																																												
5	0	0	0	0	47	0	100%																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	12.9%	0.0%	0.0%																																																																																																																												
6	0	0	0	0	0	82	100%																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	0.0%	22.5%	0.0%																																																																																																																												
7	0	0	0	0	0	0	100%																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																												
	100%	100%	100%	100%	100%	100%	100%																																																																																																																												
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%																																																																																																																												

En las tablas se presentan las matrices de confusión y las curvas ROC (Receiver Operating Characteristic) correspondientes a cada una de las configuraciones consideradas en el estudio. Solo se incluyen las matrices/para el entrenamiento completo, es decir, se trata de los resultados correspondientes a la suma de los casos de entrenamiento, test y validación.

Las matrices de confusión indican el número de aciertos y fallos que ha habido en el entrenamiento, para ello, en la matriz se generarán tantas filas y columnas como salidas de la clasificación existan (en este caso, las 7 zonas definidas en el buque). En la matriz se relaciona el output determinado por la red frente al objetivo establecido (al tratarse de un entrenamiento supervisado). De este modo, los aciertos se muestran en la diagonal de la matriz (casillas verdes) y los fallos en el resto de los elementos de la matriz (casillas rojas). Las gráficas ROC, indican la relación entre la tasa de verdaderos positivos y la tasa de falsos positivos de la red.

Como se puede ver en las tablas 5-17, 5-18 y 5-19, los resultados obtenidos empleando el método de Scalated Conjugated Gradient no fueron demasiado buenos. Se trata de un método de entrenamiento que requiere un coste computacional muy bajo, pero que en ocasiones no permite alcanzar unos resultados aceptables. En cambio, los resultados obtenidos empleando el algoritmo de Levenberg-Marquard y, sobre todo, los resultados obtenidos empleando la Regularización Bayesiana ofrecieron unos resultados realmente buenos, incluso empleando un número bajo de neuronas. En el caso de la regulación bayesiana se ha logrado un acierto de un 100% en el entrenamiento realizado para un número de capas igual a 20 y a 50, si bien, ese acierto se penaliza con un elevado tiempo de entrenamiento, sobre todo en el caso en el que haya que emplear 50 neuronas, cuando el tiempo de entrenamiento de la red alcanzó las 24h (en un pc convencional).

Las principales conclusiones del entrenamiento de la red neuronal para localización de fallos en tuberías a partir de las mediciones de los datos de los sensores de presión del sistema CI por agua salada se pueden resumir en lo siguiente:

- Se ha llevado a cabo el desarrollo de una red neuronal que permita la localización de fallos en el sistema CI del buque.

- Cada una de las configuraciones de la red se ha entrenado empleando los datos disponibles de aperturas de consumidores/simulación de rotura de tuberías, medidos en las pruebas realizadas a bordo y que se emplearon para el entrenamiento del sistema de detección de fallos.
- Debido a la disponibilidad de un número limitado de datos, se optó por dividir el buque en zonas y etiquetar cada una de las aperturas de consumidores/roturas de tuberías indicando en qué zona de las definidas se encuentra. Tratar de obtener un sistema que permita mayor precisión en la detección de fallos pasará por la obtención de una cantidad importante de datos que permitan entrenar el algoritmo basado en redes neuronales.
- Debido a que se realizó un etiquetado por zonas, se optó por emplear una red de clasificación (*patternnet* en MATLAB). Se definieron varias configuraciones diferentes para la RN, con el objetivo de determinar qué algoritmo de entrenamiento, y qué estructura de la red (número de neuronas) arroja mejores resultados. Se probaron los algoritmos de entrenamiento disponibles en MATLAB para las redes *patternnet* (Scalated Conjugated Gradient, Levenberg-Marquard y Regularización Bayesiana) y se entrenaron redes de 1 capa oculta y con 5, 10, 20 y 50 neuronas en esa capa oculta.
- Los mejores resultados se obtuvieron empleando como función de entrenamiento la Regularización Bayesiana. El algoritmo de Levenberg-Marquard también arrojó muy buenos resultados (en ambos casos no fue necesario emplear un elevado número de neuronas para obtener buenos resultados), mientras que los peores se obtuvieron empleando el método de Scalated Conjugated Gradient. Es importante destacar que las conclusiones de los resultados obtenidos se han de considerar siempre desde la premisa de que el entrenamiento se llevó a cabo empleando un número bajo de casos de entrenamiento y la identificación se ha realizado para zonas relativamente amplias del buque.

## 5.10. IMPLEMENTACIÓN DE SISTEMAS BASADOS EN RN EN EL BUQUE

Si se considera que este tipo de sistemas irá integrado en el entorno del gemelo digital del buque, la implantación dependerá de la plataforma en la

que se base el gemelo digital. En general, existen dos posibilidades para la ejecución de un sistema basado en redes neuronales:

- El modelo podrá ejecutarse en la herramienta nativa en la que se ha desarrollado (en este caso, Matlab). Esta alternativa implica que será necesario disponer del software necesario para su ejecución en el buque. Esto puede ser una importante limitación, ya que no siempre será posible llevar a cabo la instalación de determinadas herramientas en el buque. Además, se ha de tener en cuenta que podrán existir modelos desarrollados en diferentes herramientas, lo cual implicaría la instalación de una cantidad, en algunos casos elevada, de software en el buque, además de la conexión de esas herramientas que contengan los modelos con la herramienta de gemelo digital (un proceso que en general, no será trivial).
- De acuerdo con lo expuesto en el capítulo 4 de esta tesis doctoral, se considerará la exportación del modelo a formato FMU, el cual, en general, será compatible con la mayoría de las plataformas que dan soporte al gemelo digital.

En las primeras fases de la actuación, se desarrollaron pruebas para la implementación de este sistema en la plataforma IoT que en aquel momento se estaba probando, Thingworx. Se diseñaron algunas de las ventanas para un posible interfaz gráfico.

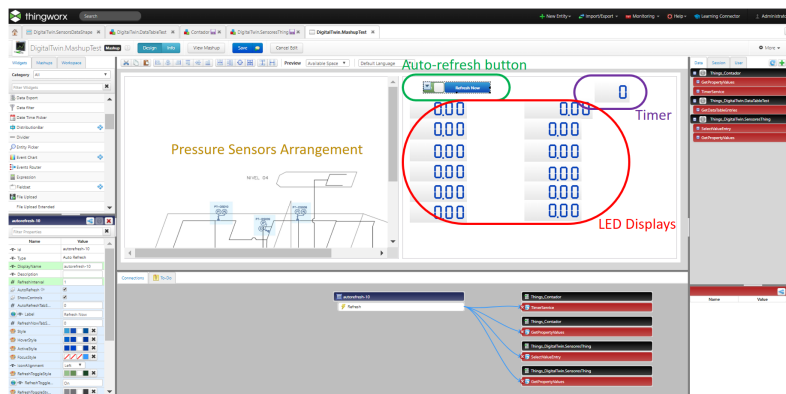
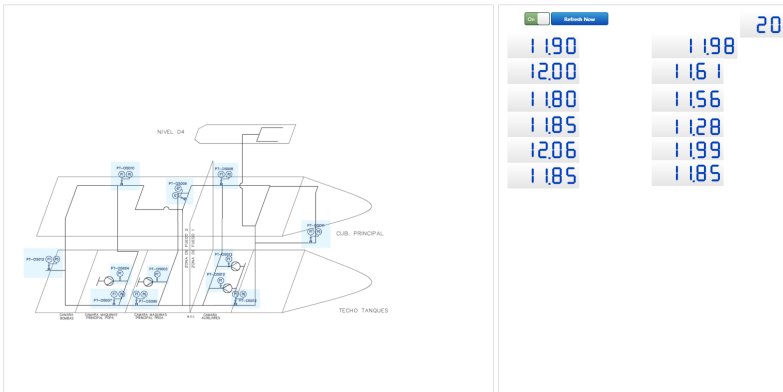
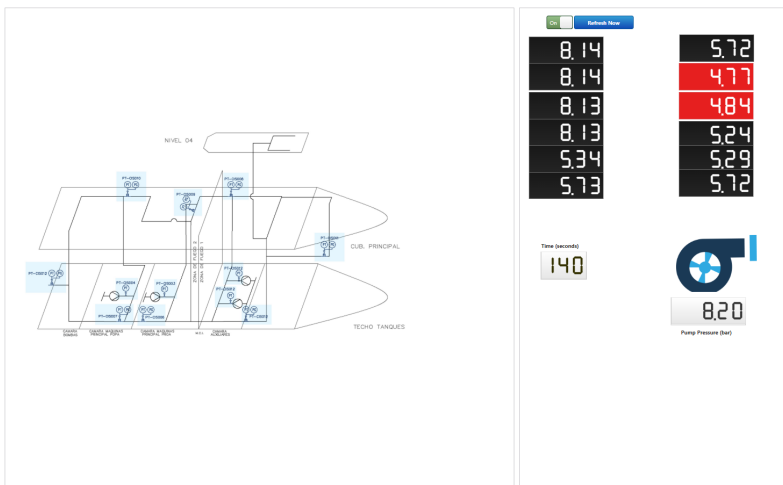


Imagen 5-32: Desarrollo de ventanas para interfaces (mashups) en Thingworx



**Imagen 5-33: Pruebas de diseño de interfaces para monitorización del sistema (funcionamiento real)**



**Imagen 5-34: Pruebas de diseño de interfaces para monitorización del sistema (fallo del sistema - alarma)**

La conexión con el sistema de control del buque, para la obtención de los datos reales de los sensores de presión del contraincendios, y la conexión con la ejecución de los modelos de detección y localización de fallos desde la plataforma Thingworx se realizará de forma análoga a lo planteado en el capítulo 4 (para la arquitectura centralizada en la plataforma de IoT). El uso de API-Rest permitirá hacer la petición al sistema de control del buque, que devolverá las medidas de los sensores de presión del sistema, y a continuación se enviarán esas medidas al algoritmo basado en redes neuronales, previamente

exportado a FMU, y que se ejecutará en el entorno del gemelo digital empleando la librería de Python para simulación de FMUs: PyFMI.

### 5.11. CONCLUSIONES

En este capítulo se ha tratado de validar el uso de algoritmos basados en inteligencia artificial para su aplicación en el entorno del gemelo digital de sistemas del buque. En concreto, se ha desarrollado un modelo basado en redes neuronales para su uso en detección y localización de fallos en el sistema contra incendios por agua salada del buque el cual se alimenta por los datos de presión que ofrece la red de sensores con la que cuenta el sistema.

Para poder entrenar la red neuronal se realizaron pruebas a bordo de un buque, a fin de obtener datos reales de funcionamiento normal del sistema. Además, se forzaron eventos que permitiesen obtener una respuesta similar a la producida por la rotura y colapso de un tramo de tubería. En las pruebas a bordo del buque realizadas en la primera fase del trabajo, se obtuvieron datos con una elevada frecuencia de muestreo (50 Hz) pero de baja resolución (décima de bar), que no pudieron emplearse en el entrenamiento. Por ello, el diseño inicial de la arquitectura de la red (y entrenamiento inicial) se llevó a cabo empleando datos simulados de un modelo de comportamiento que se desarrolló para el sistema.

Esta red fue re-entrenada posteriormente con los datos reales obtenidos en las pruebas realizadas en la segunda fase del trabajo. Estas pruebas se desarrollaron en un buque más antiguo que las primeras, por lo que en este caso no se pudo alcanzar una frecuencia de muestreo como en la primera fase (de 50 Hz), en este caso no se alcanzaron frecuencias por encima de los 10 Hz. Si bien, la resolución de la señal fue de milésima de bar, por lo que, pese a la limitación en la frecuencia, los datos obtenidos fueron válidos.

Los resultados obtenidos en esta etapa de re-entreno de la red fueron satisfactorios, teniendo en cuenta que los datos reales de funcionamiento normal (y de simulación de rotura de tubería) para entrenamiento fueron limitados. Se empleó una red de ajuste, entrenada con regularización bayesiana. Se probaron diferentes configuraciones, desde 1 capa a 10 y de 5 a 50 neuronas. Se observó que los mejores resultados (menor error en el entrenamiento) se obtienen para configuraciones de entre 4 y 7 capas ocultas. Además. Los mejores

resultados (para cualquier número de capas) se obtuvieron para 5 o 10 neuronas por capa oculta.

Además, los datos recogidos de los sensores de presión en esta segunda fase permitieron iniciar el desarrollo de un modelo para la localización de los fallos en el sistema. Para ello, se etiquetaron los diferentes eventos recogidos en las pruebas. Debido a la poca densidad de datos, se optó por dividir el buque en 7 zonas y asignar los eventos a las zonas en las que se produjeron. En este caso se optó por emplear una red de clasificación. Se consideraron diferentes configuraciones, en este caso siempre con 1 capa oculta, y con neuronas de 5 a 50. Además, se compararon los resultados empleando diferentes algoritmos de entrenamiento. Los mejores resultados se obtuvieron para la regularización bayesiana (aunque se trata de la función de entrenamiento que más costo computacional genera). El entrenamiento empleando Levenberg-Marquard también arrojó buenos resultados. Se comprobó que en ningún caso es necesario ir a más de 10 neuronas en la capa oculta para obtener resultados satisfactorios.

Además, se han de realizar los siguientes comentarios sobre la adquisición de datos:

- En general, los buques no presentan una frecuencia de registro de las señales de presión del sistema que permitan alimentar un modelo de este tipo (dado que, al buque, hasta ahora, no le interesaba recoger esas señales a tan alta frecuencia). Además, tampoco interesan señales de presión del sistema contraincendios (que trabaja, dependiendo del buque, entre 8 y 12 bar) con una elevada resolución. Pero, aunque los buques en los que han realizado pruebas no disponían, a priori de las señales configuradas para ser empleadas en el modelo, fueron fácilmente modificables (en el caso del buque más antiguo, fue necesario actuar sobre los RTU, mientras que en el buque más actual solo fue necesario actuar sobre el sistema de control).
- En ambos casos los datos se descargaron mediante un registrador de datos programado para tal fin, en uno de los casos, e instalado en una raspberry que se conectó a la red del sistema de control del buque, y en el otro caso, el programa se instaló en un PC (y fue el PC el que se



conectó a la red del sistema de control del buque, para la adquisición de los datos).

Como principales tareas para trabajo futuro enmarcadas en el desarrollo de modelos basados en IA en el entorno del gemelo digital del buque, se debe de destacar lo siguiente:

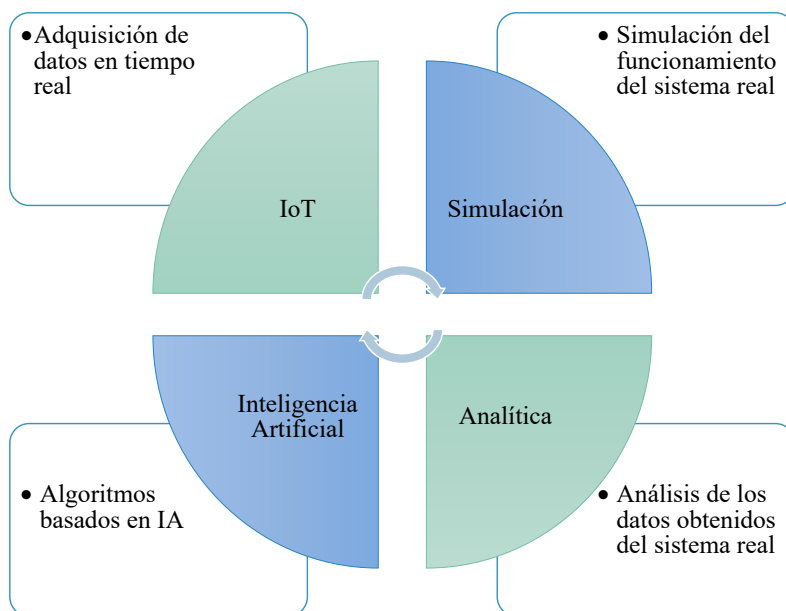
- Realización de nuevas pruebas a bordo (si es posible, pruebas de larga duración) que permitan recoger un volumen importante de datos con el que refinar el modelo para detección de fallos y que, además, permita aumentar la precisión en el modelo para localización de fallos.
- Realización de pruebas de ejecución de los modelos a bordo y en tiempo real. Valorando la posibilidad de emplear los modelos en formato FMU. Integración del modelo en el gemelo digital del buque.

## 6. CONCLUSIONES Y TRABAJO FUTURO

---

En esta tesis doctoral se ha planteado una metodología mediante la cual se busca marcar el camino a seguir en la implantación de gemelos digitales en el buque.

Esta metodología se ha basado en establecer una arquitectura para el gemelo digital del buque centrada en 4 pilares fundamentales, las 4 tecnologías habilitadoras que permiten el desarrollo de gemelos digitales, y que son (ver imagen 6-1): IoT, simulación, inteligencia artificial y analítica de datos.



**Imagen 6-1: Tecnologías habilitadoras y funcionalidades principales de un gemelo digital**

En paralelo a esos 4 pilares fundamentales, se establecen las principales funcionalidades que se han definido para poder considerar un gemelo digital:

- La adquisición de datos reales del sistema en tiempo real.
- Simulación del funcionamiento del sistema físico.

- Análisis de los datos obtenidos del sistema real.
- Algoritmos basados en inteligencia artificial que permitan que el sistema aprenda a partir de su funcionamiento, sirvan como soporte a la toma de decisiones y permitan avanzar hacia un gemelo digital inteligente del buque.

Se trató de validar la metodología planteada mediante la aplicación práctica a un sistema real, y en un entorno similar al que se puede presentar en un buque, empleando para ello las instalaciones (demostrador) que Navantia diseñó y construyó para la realización de pruebas piloto relacionados con el gemelo digital.

Además, se evaluó la implementación de algoritmos basados en inteligencia artificial para la detección de fallos en sistemas del buque. En concreto, se implantó un sistema de aprendizaje basado en inteligencia artificial para la detección y localización de fallos en el sistema contraincendios por agua salada del buque, basado en la lectura de los sensores de presión del sistema.

Se resumen a continuación las principales conclusiones y el trabajo futuro en cada una de las líneas de trabajo abiertas en los dos casos de uso planteados.

## **6.1. IMPLANTACIÓN DE LA METODOLOGÍA PROPUESTA EN UN ENTORNO REAL**

Se analizaron los componentes definidos en la arquitectura planteada en la metodología, en especial, los modelos de simulación y su implantación y ejecución en el entorno del gemelo digital. También se analizó la adquisición de datos desde el sistema real hacia el gemelo digital y el almacenamiento de información en el gemelo.

Se validaron dos alternativas diferentes para el despliegue del gemelo digital en el buque: un despliegue del gemelo digital centralizado en una plataforma para IoT, y un segundo despliegue del gemelo digital descentralizado.

En la primera alternativa planteada se analizó el despliegue de un gemelo digital del buque centralizado en una plataforma para IoT, incluyendo lo siguiente:

- Se validó una alternativa para la adquisición de datos desde el sistema real hacia el gemelo digital.

- Se desarrollaron modelos de simulación del sistema real en una herramienta para simulación de sistemas multifísicos: OpenModelica. Esos modelos exportados al estándar FMI, se ejecutaron en el entorno del gemelo digital.
- Se analizó el almacenamiento de datos históricos en la plataforma IoT.
- Se validó la conexión de la plataforma IoT para gemelo digital con el gestor del ciclo de vida del buque y con el ERP.

En la segunda alternativa planteada, se analizó el despliegue para implementación del gemelo digital de un sistema HVAC, basado en una arquitectura no centralizada en una plataforma de IoT, en concreto:

- Se desarrollaron modelos de simulación del sistema real en una herramienta de simulación de sistemas multifísicos: AMESIM. Los modelos exportados a formato FMU (estándar FMI) se ejecutaron en la capa de edge computing del gemelo digital.
- Se validó el uso de los modelos de simulación para la detección de fallos en el sistema real, incluyendo la visualización de los datos reales frente a los simulados, así como avisos de alarmas por desviación de los datos reales respecto a los del modelo.

El trabajo futuro planteado como resultado de este caso se uso, se centrará en el desarrollo de modelos y su implantación en los entornos de gemelo digital en el buque. Debido a que, en la última fase del trabajo, se recogió un importante volumen de datos históricos, se plantea lo siguiente:

- Ampliar el desarrollo realizado de modelos de simulación orientados a gemelos digitales mediante el avance en el análisis de la co-simulación de FMUs, entendiendo que la co-simulación será un elemento crítico a la hora de implantar modelos de simulación en los gemelos digitales de los sistemas del buque. Se plantearán estudios de compatibilidad e integración de FMUs y de escalabilidad, para tratar de validar hasta que nivel de detalle en modelos se puede alcanzar en un desarrollo lo más parecido a un buque real.

- Llevar a cabo el desarrollo de modelos de orden reducido basados en los datos reales de funcionamiento del sistema, ya que, en el caso de uso planteado, no se llegaron a desarrollar modelos de orden reducido por no disponer de la herramienta necesaria.
- Los históricos recogidos, contienen un volumen considerable de datos de funcionamiento normal del sistema, así como de 2 fallos: por un lado, la obstrucción del filtro de entrada del aire en el unit cooler y, por otro lado, una situación en la que la correa que comunica el motor del ventilador con el ventilador está aflojada. Estos datos podrán ser empleados para analizar el uso de algoritmos de aprendizaje en este entorno.

## **6.2. DESARROLLO DE ALGORITMOS BASADOS EN INTELIGENCIA ARTIFICIAL EN EL GEMELO DIGITAL PARA DETECCIÓN Y LOCALIZACIÓN DE FALLOS EN EL BUQUE**

En lo referente a la última parte de esta tesis doctoral, centrada en la validación del uso de algoritmos basados en inteligencia artificial para la detección y localización de fallos en sistemas del buque, y aplicado al sistema contraincendios de agua salada, es necesario hacer las siguientes consideraciones:

- Se buscaba obtener un sistema de detección y localización de fallos en el sistema contraincendios del buque, basado en los datos de los sensores de presión existentes en el sistema, y empleando algoritmos basados en redes neuronales, llevando a cabo un entrenamiento supervisado.
- Se llevaron a cabo 2 pruebas principales de toma de datos para entrenamiento del sistema. Los datos obtenidos en las primeras pruebas no fueron satisfactorios, ya que no fue posible identificar los fallos que se simularon (principalmente por la baja resolución de las señales de presión que solo alcanzaban la décima de bar. Por ese mismo motivo, la caracterización del funcionamiento normal también fue muy limitada.
- Debido a lo anterior, se propuso la realización de nuevas pruebas en un buque. La situación derivada de la pandemia COVID-19,

retrasó la realización de las pruebas, y cuando se realizaron solo fue posible llevarlas a cabo en un buque diferente al empleado en las primeras pruebas.

- La toma de datos en esta segunda prueba ha permitido validar el modelo de redes neuronales desarrollado, basado en un número limitado de consumidores del sistema y de simulación de roturas de tuberías.
- En el caso de la localización de fallos, se consiguió localizar la zona del buque en la que se produjo un fallo (o se abrió un consumidor del sistema).

El trabajo futuro estará orientado a:

- Ampliación de pruebas para obtención de más datos para ampliar la detección y localización de fallos (acotar más la localización en lugar de hacerlo por zonas). Se ha tratado el tema con Navantia, y por su lado están muy interesados en continuar con este desarrollo en el marco de lo que se ha denominado "asistente en averías".
- Validación del sistema en tiempo real. Se tratará de implementar el sistema en un buque real, con el objetivo de validar su funcionamiento, y al mismo tiempo permitir que el sistema siga aprendiendo con los datos que va recogiendo.

## 7. REFERENCIAS

---

- ABB. (2010). *Communication Protocol MODBUS-TCP Quick Overview*.
- Abdulshaheed, A., Mustapha, F., & Ghavamian, A. (2017). A pressure-based method for monitoring leaks in a pipe distribution system: A Review. *Renewable and Sustainable Energy Reviews*, 69(August 2016), 902–911. <https://doi.org/10.1016/j.rser.2016.08.024>
- Adamenko, D., Kunnen, S., Pluhnau, R., Loibl, A., & Nagarajah, A. (2020). Review and comparison of the methods of designing the Digital Twin. *Procedia CIRP*, 91, 27–32. <https://doi.org/10.1016/j.procir.2020.02.146>
- AIMEN, Government, R., Industry, G. N., Agency, G. I., Financing, A., Shipbuilding, C., Galicia, C. I. S., & Ambit, S. (2020). *Shipbuilding 4.0 - Dynamization of innovation capacity of galician marine sector*. <https://www.aimen.es/proyectos/shipbuilding-4-0-dinamizacion-de-la-capacidad-innovadora-del-sector-naval-gallego>
- Alam, K. M., & El Saddik, A. (2017). C2PS: A digital twin architecture reference model for the cloud-based cyber-physical systems. *IEEE Access*, 5(January), 2050–2062. <https://doi.org/10.1109/ACCESS.2017.2657006>
- Aljukic, A. (2017). *Industry 4.0: an Australian perspective*. March, 19 p. <https://www.standards.org.au/StandardAU/Media/SA-Archive/OurOrganisation/News/Documents/Industry-4-0-Recommendations-Report.pdf>
- Andersson, C., Akesson, J., & Fuhrer, C. (2016). PyFMI: A Python Package for Simulation of Coupled Dynamic Models with the Functional Mock-up Interface. *Technical Report in Mathematical Sciences*, 2, 1–40.
- Asbjørnslett, B. E., Erikstad, S. O., Rølvåg, T., & Borgersen, M. (2018). *Report on the use of digital twins in engineering education*.
- Ashtari Talkhestani, B., Jung, T., Lindemann, B., Sahlab, N., Jazdi, N., Schloegl, W., & Weyrich, M. (2019). An architecture of an Intelligent Digital Twin in a Cyber-Physical Production System. *At-Automatisierungstechnik*, 67(9), 762–782. <https://doi.org/10.1515/auto-2019-0039>
- Astarloa, A., & Rodríguez, M. (2018). Seguridad y Time Sensitive. *InfoPLC++*, 38–43.

- Australia Germany Advisory Group. (2015). *Collaboration, Innovation and Opportunity - Report of the Australia-Germany Advisory Group*. November. <https://dfat.gov.au/about-us/publications/international-relations/Documents/australia-germany-advisory-group.pdf>
- Azarmipour, M., Elfaham, H., Gries, C., & Epple, U. (2019). PLC 4.0: A Control System for Industry 4.0. *IECON Proceedings (Industrial Electronics Conference)*, 2019-October, 5513–5518. <https://doi.org/10.1109/IECON.2019.8927026>
- Belhadi, A., Zkik, K., Cherrafi, A., Yusof, S. M., & El fezazi, S. (2019). Understanding Big Data Analytics for Manufacturing Processes: Insights from Literature Review and Multiple Case Studies. *Computers and Industrial Engineering*, 137(September), 106099. <https://doi.org/10.1016/j.cie.2019.106099>
- Berardi, L., Laucelli, D., Ugarelli, R., & Giustolisi, O. (2015). Leakage management: Planning remote real time controlled pressure reduction in Opegård municipality. *Procedia Engineering*, 119(1), 72–81. <https://doi.org/10.1016/j.proeng.2015.08.855>
- Bermúdez, J.-R., López-Estrada, F.-R., Besançon, G., Valencia-Palomo, G., Torres, L., & Hernández, H.-R. (2018). Modeling and Simulation of a Hydraulic Network for Leak Diagnosis. *Mathematical and Computational Applications*, 23(4), 70. <https://doi.org/10.3390/mca23040070>
- Bertsch, C., Ahle, E., & Schulmeister, U. (2014). The Functional Mockup Interface - seen from an industrial perspective. *Proceedings of the 10th International Modelica Conference, March 10-12, 2014, Lund, Sweden*, 96(October), 27–33. <https://doi.org/10.3384/ecp1409627>
- Bilic, D., Brosse, E., Sadovykh, A., Truscan, D., Bruneliere, H., & Ryssel, U. (2019). An integrated model-based tool chain for managing variability in complex system design. *Proceedings - 2019 ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems Companion, MODELS-C 2019*, 288–293. <https://doi.org/10.1109/MODELS-C.2019.00045>
- Bleakley, G. (2020). *The role of Simulation and AI in the implementation of a Digital Twin*.
- Breu, F., Guggenbichler, S., & Wollmann, J. (2008). How to Make your Developers Rest. In *Vasa*.



- <http://medcontent.metapress.com/index/A65RM03P4874243N.pdf>
- Bruun, K. (2018). Digital Twins for Blue Denmark Danish Maritime Authority. In *DIGITAL TWIN REPORT FOR DMA: Vol. A* (Issue 6). [www.dnvgl.com](http://www.dnvgl.com)
- Bruun, K., Levi, L., Jamt, K., & Husteli, N. (2016). Digital twins for design, testing and verification throughout a vessel's life cycle. *COMPIT2016*.
- Buchheit, M., Ferraro, A., Lim, C., Lin, S.-W., Morrish, J., & Zarkout, B. (2020). Digital Transformation in Industry White Paper. *Industrial Internet Consortium*.
- Burke, T. (2016). OPC Unified Architecture - Interoperability for Industrie 4.0 and the Internet of Things. *OPC Unified Architecture: Interoperability for Industrie 4.0 and the Internet of Things*, 16–24.
- BWMI. (2017). Maritime Agenda 2025. The future of Germany as a maritime industry hub. *The Federal Ministry for Economic Affairs and Energy (BMWi)*.
- Cabos, C., & Rostock, C. (2018). Digital Model or Digital Twin? *Proceedings of the 2018 Conference on Computer Applications and Information Technology in the Maritime Industries (COMPIT)*.
- Cabos, Christian, & Rostock, C. (2018). Digital Model or Digital Twin? *17th Conference on Computer Applications and Information Technology in the Maritime Industries (COMPIT)*, 403–411.
- CATIA-Systems. (2017). *FMPy - FMU container*. [https://github.com/CATIA-Systems/FMPy/blob/master/build\\_fmcontainer.py](https://github.com/CATIA-Systems/FMPy/blob/master/build_fmcontainer.py)
- Chan, B. J., & Kumar, M. (2019). *Real-Time Digital Twins of At-Sea Ships Speed Warfighting Capability to the Fleet*. <https://www.doncio.navy.mil/CHIPS/ArticleDetails.aspx?ID=12106>
- Chombart, P. (2012). Multidisciplinary modelling and simulation speeds development of automotive systems and software. *ITEA2 Innovation Report*, 33(0), 52–54. [https://itea3.org/project/result/download/6202/MODELISAR Innovation Report.pdf](https://itea3.org/project/result/download/6202/MODELISAR%20Innovation%20Report.pdf)
- Chukalov, K. (2017). Horizontal and Vertical Integration, As a Requirement for Cyber-Physical Systems in the Context of Industry 4.0. *International Scientific Journal "Industry 4.0,"* 2(4), 155–157. <http://www.plattform-i40.de/sites/>

- CISCO. (2017). *Time-Sensitive Networking : A Technical Introduction*.
- CISCO. (2020). *Cisco Annual Internet Report (2018–2023)*.  
<https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.pdf>
- Coraddu, A., Oneto, L., Baldi, F., Cipollini, F., Atlar, M., & Savio, S. (2019). Data-driven ship digital twin for estimating the speed loss caused by the marine fouling. *Ocean Engineering*, 186(January), 106063.  
<https://doi.org/10.1016/j.oceaneng.2019.05.045>
- DAMEN. (2018). *DAMEN: Research, Development & Innovation*. 1–28.  
[www.alticelabs.com](http://www.alticelabs.com)
- DAMEN. (2019). *Smart sensor mapping breaks new ground in maritime remote monitoring*. Damen Magazine.  
<https://www.damen.com/en/magazines/2019/02/smart-sensor-mapping-breaks-new-ground-in-maritime-remote-monitoring>
- Datta, S., & Sarkar, S. (2016). A review on different pipeline fault detection methods. *Journal of Loss Prevention in the Process Industries*, 41, 97–106.  
<https://doi.org/10.1016/j.jlp.2016.03.010>
- Delligatti, L. (2014). *SysML Distilled*.
- Deloitte. (2017). *Industry 4.0: The birth of the smart factory*.
- Dertien, S., Lang, J., & Immerman, D. (2019). *Digital Twin: A primer for industrial enterprise*.
- Diaz-Casas, V., Vilar-Montesinos, M., Munin-Doce, A., Diaz-Bouza, M.-A., Trueba-Martinez, P., & Ferreno-Gonzalez, S. (2018). Industrial Internet of Things (IIOT): The digital challenge of Shipyard 4.0. *Ingeniería Naval*, 965.
- Dilberoglu, U. M., Gharehpapagh, B., Yaman, U., & Dolen, M. (2017). The Role of Additive Manufacturing in the Era of Industry 4.0. *Procedia Manufacturing*, 11(June), 545–554.  
<https://doi.org/10.1016/j.promfg.2017.07.148>
- Dorwart, J. (2017). *Newport News Shipbuilding Digital Journey : Drawings to 4GD based Visual Instructions Newport News Shipbuilding*. April.
- DSpace. (2021). *Use Case : Integrating Functional Mock-up Units for HIL Simulation*. SCALEXIO Support for FMI. Use Case: FMUs for PC-Based

Simulation.

[https://www.dspace.com/en/pub/home/products/hw/simulator\\_hardware/scalexio/scalexio\\_fmi/hil\\_fmi.cfm](https://www.dspace.com/en/pub/home/products/hw/simulator_hardware/scalexio/scalexio_fmi/hil_fmi.cfm)

Erikstad, S. O. (2018). Design patterns for digital twin solutions in marine systems. *Comput* 2018, June, 354–363. <https://www.researchgate.net/publication/325871050>

Erikstad, S. O., & Ove, S. (2017). Merging Physics, Big Data Analytics and Simulation for the Next-Generation Digital Twins. *High-Performance Marine Vehicles*, October, 11–13. <https://www.predix.io>.

*Estrategias para el fomento de la Introducción*. (2020).

ETAS. (2016). *LABCAR-OPERATOR V5.4.1 User's Guide*.

European Defence Agency. (2020). *Study on Industrial and Technological Competences in the Naval Sector: Vol. 14.CPS.OP.* (Issue 14).

Ferguson, G. (2020). *Blog BAE Systems Australia and Australian research partners to develop shipbuilding smarts*. <https://rumourcontrol.com.au/bae-systems-australia-and-australian-research-partners-to-develop-shipbuilding-smarts/>

Ferrer, B. R., Mohammed, W. M., Martinez Lastra, J. L., Villalonga, A., Beruvides, G., Castano, F., & Haber, R. E. (2018). Towards the Adoption of Cyber-Physical Systems of Systems Paradigm in Smart Manufacturing Environments. *Proceedings - IEEE 16th International Conference on Industrial Informatics, INDIN 2018*, 792–799. <https://doi.org/10.1109/INDIN.2018.8472061>

Foresee, F. D., & Hagan, M. T. (1930). GAUSS-NEWTON APPROXIMATION TO BAYESIAN LEARNING. *Network*, 1930–1935.

Fritzson, P. (2011). Functional Mockup Interface (FMI) A General Standard for Model Exchange and Simulator Coupling. *Linköping University*, 1–32.

Fu, T. C. (2017). Navy Platform Digital Twin. In *Navy Digital Twin (NDT)*.

Fundación Telefónica. (2015). La transformación digital de la industria española. Informe preliminar. *Industria Conectada 4.0*, 120.

Gallegos Valera, M. C. (2013). Introducción a los Servicios Web RESTful. In *Los servicios web XML*. <http://www.jtech.ua.es/j2ee/restringido/cw/sesion11-apuntes.pdf>

- Gao, Z., Wanyama, T., Singh, I., Gadhri, A., & Schmidt, R. (2020). From industry 4.0 to robotics 4.0 - A conceptual framework for collaborative and intelligent robotic systems. *Procedia Manufacturing*, 46(2019), 591–599. <https://doi.org/10.1016/j.promfg.2020.03.085>
- García González, P. (2018). *Gemelo digital. Concepto y aplicación a la armada española*. <http://oa.upm.es/53240/>
- Gavin, H. P. (2019). The Levenberg-Marquardt Algorithm For Nonlinear Least Squares Curve-Fitting Problems. *Duke University*, 1–19. <http://people.duke.edu/~hpgavin/ce281/lm.pdf>
- General Electric. (2018). PREDIX: The application platform for digital industrial solutions. In *General-Electric*. <https://www.ge.com/digital/sites/default/files/Predix-from-GE-Digital-Overview-Brochure.pdf>
- General Electric Company. (2015). *General Electric Digital Twin* (Issue 2). <https://www.siemens.com/customer-magazine/en/home/industry/digitalization-in-machine-building/the-digital-twin.html>
- Ghimire, P., Zadeh, M., & Pedersen, E. (2021). Co-simulation of a marine hybrid power system for real-time virtual testing. *2021 IEEE Transportation Electrification Conference and Expo, ITEC 2021*, 1–6. <https://doi.org/10.1109/ITEC51675.2021.9490050>
- Giustolisi, O., Campisano, A., Ugarelli, R., Laucelli, D., & Berardi, L. (2015). Leakage management: WNetXL Pressure Control module. *Procedia Engineering*, 119(1), 82–90. <https://doi.org/10.1016/j.proeng.2015.08.856>
- Gkerekos, C., Theotokatos, G., Bujorianu, L. M., Boulougouris, E., Vassalos, D., Carballedo, B., & McCluskey, S. (2019). Digitalisation in the UK maritime sector: A stakeholders' pulse check. *RINA, Royal Institution of Naval Architects - International Conference on Marine Industry 4.0, Papers, November*, 63–69. <https://doi.org/10.3940/rina.mi.2019.09>
- Glaessgen, E. H., & Stargel, D. S. (2012). The digital twin paradigm for future NASA and U.S. Air force vehicles. *Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, 432938. <https://doi.org/10.2514/6.2012-1818>
- Gómez-Camperos, J. A., Espinel-Blanco, E. E., & Regino-Ubarnes, F. J. (2019).

- Diagnosis of horizontal pipe leaks using neural networks. *Journal of Physics: Conference Series*, 1388(1). <https://doi.org/10.1088/1742-6596/1388/1/012032>
- González, I., Calderón, A. J., Figueiredo, J., & Sousa, J. M. C. (2019). A literature survey on open platform communications (OPC) applied to advanced industrial environments. *Electronics (Switzerland)*, 8(5), 1–29. <https://doi.org/10.3390/electronics8050510>
- Graessler, I., & Poehler, A. (2018). Integration of a digital twin as human representation in a scheduling procedure of a cyber-physical production system. *IEEE International Conference on Industrial Engineering and Engineering Management, 2017-Decem*, 289–293. <https://doi.org/10.1109/IEEM.2017.8289898>
- Greco, L., Maresca, P., & Caja, J. (2019). Big Data and Advanced Analytics in Industry 4.0: A comparative analysis across the European Union. *Procedia Manufacturing*, 41, 383–390. <https://doi.org/10.1016/j.promfg.2019.09.023>
- Grieves, M., & Vickers, J. (2017). Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems. In *Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches* (pp. 85–113). Springer International Publishing.
- Grieves, Michael, & Vickers, J. (2016). Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems. In *Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches* (Issue August 2017). <https://doi.org/10.1007/978-3-319-38756-7>
- Harper, K. E., Ganz, C., & Harper, K. E. (2019). *Digital Twin Architecture and Standards*. November, 0–12. <https://www.iiconsortium.org/news/joi-articles/2019-November-JoI-Digital-Twin-Architecture-and-Standards.pdf>
- Hartmann, D., Herz, M., & Wever, U. (2017). Model Order Reduction A key technology for Digital Twins. In *Reduced-Order Modeling (ROM) for Simulation and Optimization* (Issue April, pp. 1–179). <https://doi.org/10.1007/978-3-319-75319-5>
- Hatledal, L. I., Skulstad, R., Li, G., Styve, A., & Zhang, H. (2020). Co-simulation as a Fundamental Technology for Twin Ships. *Modeling, Identification and Control*, 41(4), 297–311. <https://doi.org/10.4173/MIC.2020.4.2>
- Hatledal, Lars Ivar. (2021). *Doctoral thesis For Demanding Maritime Operations*

*Protocols and Standards for Simulation and Co-simulation.*

- He, R., Chen, G., Dong, C., Sun, S., & Shen, X. (2019). Data-driven digital twin technology for optimized control in process systems. *ISA Transactions*, 95, 221–234. <https://doi.org/10.1016/j.isatra.2019.05.011>
- Hermann, M., Pentek, T., & Otto, B. (2015). Design Principles for Industrie 4.0 Scenarios: A Literature Review. *Technische Universität Dortmund*, 1(1), 4–16. <https://doi.org/10.13140/RG.2.2.29269.22248>
- Hewlett Packard. (2019). *El Internet de las Cosas: El presente y el futuro*. 1–20. [https://www.roastbrief.com.mx/wp-content/uploads/2017/03/HP\\_IoT\\_Research\\_Report\\_SPANISH.pdf](https://www.roastbrief.com.mx/wp-content/uploads/2017/03/HP_IoT_Research_Report_SPANISH.pdf)
- Hexagon. (2019). Adams Real Time. Hardware-in-the-loop Solution. In *MSC Software*.
- IBM. (2020). IBM Watson IoT Platform. *Console.Bluemix.Net*, 1. [https://www.ibm.com/support/knowledgecenter/en/SSQP8H/iot/kc\\_welcome.htm%0Ahttps://www.ibm.com/ru-ru/marketplace/internet-of-things-cloud](https://www.ibm.com/support/knowledgecenter/en/SSQP8H/iot/kc_welcome.htm%0Ahttps://www.ibm.com/ru-ru/marketplace/internet-of-things-cloud)
- Immerman, D., & Lang, J. (2019). *Top Use Cases for Digital Twin Technology to Drive Digital Transformation*.
- Inside BAE Systems ' digital naval shipyard*. (n.d.). Global Defence Technology. [https://defence.nridigital.com/global\\_defence\\_technology\\_feb18/inside\\_bae\\_systems\\_digital\\_naval\\_shipyard](https://defence.nridigital.com/global_defence_technology_feb18/inside_bae_systems_digital_naval_shipyard)
- Institute for Security & Development Policy. (2018). *Made In China 2025: Backgrounder report*. June, 1–9. [www.isdp.eu](http://www.isdp.eu)
- International Maritime Organization. (2019). *IMO Action To Reduce Ghg Emissions From International Shipping*. 44(0). [http://www.imo.org/en/MediaCentre/HotTopics/Documents/IMO ACTION TO REDUCE GHG EMISSIONS FROM INTERNATIONAL SHIPPING.pdf](http://www.imo.org/en/MediaCentre/HotTopics/Documents/IMO_ACTION_TO_REDUCE_GHG_EMISSIONS_FROM_INTERNATIONAL_SHIPPING.pdf)
- Interreg-Atlantic-Area. (2019). *Joint diagnosis of the current situation of the implementation of Industry 4.0 in the naval sector in the Atlantic area*.
- Isaja, M., & Calà, A. (2020). *Blockchain as a Key Enabling Technology for Decentralized Cyber-Physical Production Systems*.

- J. McCarthy, M.L. Minsky, N. Rochester, & C.E. Shannon. (1955). *A proposal for the Dartmouth Summer Research Project on Artificial Intelligence*. 1–13.
- Jaeckle, T. (2020). *Eclipse Ditto: Digital Twins for Eclipse IoT*.
- Javadiha, M., Blesa, J., Soldevila, A., & Puig, V. (2019). Leak localization in water distribution networks using deep learning. *2019 6th International Conference on Control, Decision and Information Technologies, CoDIT 2019*, 1426–1431. <https://doi.org/10.1109/CoDIT.2019.8820627>
- Jia, Z., Ren, L., Li, H., & Sun, W. (2018). Pipeline leak localization based on FBG hoop strain sensors combined with BP neural network. *Applied Sciences (Switzerland)*, 8(2). <https://doi.org/10.3390/app8020146>
- Junghanns, A., & Blochwitz, T. (2018). 10 Years of FMI Where are we now? Where do we go? *Japanese Modelica Conference 2018*.
- Kagermann, H., Wahlster, W., & Helbig, J. (2013). Securing the future of German manufacturing industry: Recommendations for implementing the strategic initiative INDUSTRIE 4.0. *Final Report of the Industrie 4.0 Working Group, April*, 1–84.
- Kalsoom, T., Ahmed, S., Rafi-Ul-shan, P. M., Azmat, M., Akhtar, P., Pervez, Z., Imran, M. A., & Ur-Rehman, M. (2021). Impact of IoT on manufacturing industry 4.0: A new triangular systematic review. *Sustainability (Switzerland)*, 13(22), 1–22. <https://doi.org/10.3390/su132212506>
- Kang, J., Park, Y. J., Lee, J., Wang, S. H., & Eom, D. S. (2018). Novel leakage detection by ensemble CNN-SVM and graph-based localization in water distribution systems. *IEEE Transactions on Industrial Electronics*, 65(5), 4279–4289. <https://doi.org/10.1109/TIE.2017.2764861>
- Keppel. (2019). *OFFSHORE MARINE: A newsletter of Keppel Offshore & Marine*. 2, 1–28.
- KETmaritime. (2019). *Case studies on KETs marine applications. Case 1: Advanced Manufacturing Shipbuilding Applications*.
- Knebel, F. P., Wickboldt, J. A., & de Freitas, E. P. (2020). *A Cloud-Fog Computing Architecture for Real-Time Digital Twins*. <http://arxiv.org/abs/2012.06118>
- Kofman, E. (n.d.). *Introducción a Modelica*. 37. [https://www.fceia.unr.edu.ar/control/modelica/notas\\_mod.pdf](https://www.fceia.unr.edu.ar/control/modelica/notas_mod.pdf)
- Kumar Deepak. (2018). *Best Practices for Building RESTful Web Services*.

- Laurentys, C. A., Bomfim, C. H. M., Menezes, B. R., & Caminhas, W. M. (2011). Design of a pipeline leakage detection using expert system: A novel approach. *Applied Soft Computing Journal*, 11(1), 1057–1066. <https://doi.org/10.1016/j.asoc.2010.02.005>
- Lee, J., Davari, H., Singh, J., & Pandhare, V. (2018). Industrial Artificial Intelligence for industry 4.0-based manufacturing systems. *Manufacturing Letters*, 18(October), 20–23. <https://doi.org/10.1016/j.mfglet.2018.09.002>
- Lee, J., Lapira, E., Yang, S., & Kao, A. (2013). Predictive manufacturing system - Trends of next-generation production systems. In *IFAC Proceedings Volumes (IFAC-PapersOnline)* (Vol. 46, Issue 7). IFAC. <https://doi.org/10.3182/20130522-3-BR-4036.00107>
- Lee, J., Singh, J., & Azamfar, M. (2019). Industrial artificial intelligence. *ArXiv*.
- Leu, S. Sen, & Bui, Q. N. (2016). Leak Prediction Model for Water Distribution Networks Created Using a Bayesian Network Learning Approach. *Water Resources Management*, 30(8), 2719–2733. <https://doi.org/10.1007/s11269-016-1316-8>
- Lewis, G. (2018). *ABS to Pilot Bow-to-Stern Condition-Based Class for U.S. Navy's MSC*. American Bureau of Shipping. <https://ww2.eagle.org/en/news/press-room/abs-to-pilot-bow-to-stern-condition-based-class-for-us-navy-msc.html>
- Lim, M. K., Li, Y., Wang, C., & Tseng, M. (2021). A literature review of blockchain technology applications in supply chains: A comprehensive analysis of themes, methodologies and industries. *Computers & Industrial Engineering*, 154(July 2020), 107133. <https://doi.org/10.1016/j.cie.2021.107133>
- Liu, X., Zhang, T., Hu, N., Zhang, P., & Zhang, Y. (2020). The method of Internet of Things access and network communication based on MQTT. *Computer Communications*, 153(December 2019), 169–176. <https://doi.org/10.1016/j.comcom.2020.01.044>
- Lopez, J., & Caicedo, E. (2005). Entrenamiento Bayesiano para Redes Neuronales Artificiales. *Conference Paper, January*, 1–11.
- Lu, H., Iseley, T., Behbahani, S., & Fu, L. (2020). Leakage detection techniques for oil and gas pipelines: State-of-the-art. *Tunnelling and Underground*



- Space Technology*, 98(January). <https://doi.org/10.1016/j.tust.2019.103249>
- M. Rüssmann, Lorentz, M., Gerbert, P., M.Waldner, Justus, J., Engel, P., & Harnisch, M. (2015). Industry 4.0: The Future of Productivity and Growth in Manufacturing Industries. *Smart Innovation, Systems and Technologies*, 132, 205–215. [https://doi.org/10.1007/978-981-13-3384-2\\_13](https://doi.org/10.1007/978-981-13-3384-2_13)
- MacKay, D. J. C. (1992). Bayesian Interpolation. *Neural Computation*, 4(3), 415–447. <https://doi.org/10.1162/neco.1992.4.3.415>
- Madakam, S., Ramaswamy, R., & Tripathi, S. (2015). Internet of Things (IoT): A Literature Review. *Journal of Computer and Communications*, 03(05), 164–173. <https://doi.org/10.4236/jcc.2015.35021>
- Madni, A., Madni, C., & Lucero, S. (2019). Leveraging Digital Twin Technology in Model-Based Systems Engineering. *Systems*, 7(1), 7. <https://doi.org/10.3390/systems7010007>
- Magargle, R., Johnson, L., Mandloi, P., Davoudabadi, P., Kesarkar, O., Krishnaswamy, S., Batteh, J., & Pitschaikani, A. (2017). A Simulation-Based Digital Twin for Model-Driven Health Monitoring and Predictive Maintenance of an Automotive Braking System. *Proceedings of the 12th International Modelica Conference, Prague, Czech Republic, May 15-17, 2017*, 132, 35–46. <https://doi.org/10.3384/ecp1713235>
- Malakuti, S., van Schalkwyk, P., Boss, B., Ram Sastry, C., Runkana, V., Lin, S.-W., Rix, S., Green, G., Baechle, K., & Varan Nath, C. (2020). Digital twins for industrial applications. *IIC Journal of Innovation*, 1–19.
- Malik, P. K., Sharma, R., Singh, R., Gehlot, A., Satapathy, S. C., Alnumay, W. S., Pelusi, D., Ghosh, U., & Nayak, J. (2021). Industrial Internet of Things and its Applications in Industry 4.0: State of The Art. *Computer Communications*, 166(August 2020), 125–139. <https://doi.org/10.1016/j.comcom.2020.11.016>
- Masood, T., & Egger, J. (2019). Augmented reality in support of Industry 4.0—Implementation challenges and success factors. *Robotics and Computer-Integrated Manufacturing*, 58(March), 181–195. <https://doi.org/10.1016/j.rcim.2019.02.003>
- Mengist, A., Asghar, A., Pop, A., Fritzsion, P., Braun, W., Siemers, A., & Fritzsion, D. (2015). An Open-Source Graphical Composite Modeling Editor and Simulation Tool Based on FMI and TLM Co-Simulation. *Proceedings of the 11th International Modelica Conference, Versailles, France*,

- September 21-23, 2015, 118, 181–188. <https://doi.org/10.3384/ecp15118181>
- Meseguer, J., Mirats-Tur, J. M., Cembrano, G., & Puig, V. (2015). Model-based monitoring techniques for leakage localization in distribution water networks. *Procedia Engineering*, 119(1), 1399–1408. <https://doi.org/10.1016/j.proeng.2015.08.1000>
- Ministerio de Industria, C. y T. (2019). *Agenda Sectorial de la Instrustria Naval*. 126.
- Modeliosoft Consulting Team. (2009). *Practical Use Case Guide*.
- Modelon AB. (2018). *JModelica . org User Guide*.
- Morais, D. (n.d.). *NAVIGATING THE TRANSFORMATIONS COMING TO SHIPBUILDING*.
- Navantia. (2020). *Guía de orientación sobre Gemelo digital*. <https://www.plm.automation.siemens.com/global/es/our-story/glossary/digital-twin/24465>
- NAVWAR. (2019). *NAVWAR Completes First Digital Model of Systems on USS Abraham Lincoln*. <https://www.navy.mil/Press-Office/Press-Releases/display-pressreleases/Article/2237404/navwar-completes-first-digital-model-of-systems-on-uss-abraham-lincoln/>
- Negri, E., Fumagalli, L., Cimino, C., & MacChi, M. (2019). FMU-supported simulation for CPS digital twin. *Procedia Manufacturing*, 28, 201–206. <https://doi.org/10.1016/j.promfg.2018.12.033>
- Negri, E., Fumagalli, L., & Macchi, M. (2017). A Review of the Roles of Digital Twin in CPS-based Production Systems. *Procedia Manufacturing*, 11(June), 939–948. <https://doi.org/10.1016/j.promfg.2017.07.198>
- Nishioka, Y. (2016). The Industrial Value Chain Initiative. *Forum Industrie 4.0 Meets the Industrial Internet*, April. [https://iv-i.org/en/docs/doc\\_160428\\_hannover.pdf](https://iv-i.org/en/docs/doc_160428_hannover.pdf)
- NISS. (2016). *Blueprint 2050: The maritime world beyond the horizon*. *Stichting Ondersteuningsfonds Nationaal Instituut Voor de Scheepvaart En Scheepsbouw*.
- O'Rourke, R. (2018). Navy force structure and shipbuilding plans: Background and issues for congress. *Defense Economics: Recent Progress and Remaining*

*Challenges*, 149–268.

- Ochel, L., Braun, R., Thiele, B., Asghar, A., Buffoni, L., Eek, M., Fritzson, P., Fritzson, D., Horkeby, S., Hällquist, R., Kinnander, Å., Palanisamy, A., Pop, A., & Sjölund, M. (2019). OMSimulator - Integrated FMI and TLM-based Co-simulation with Composite Model Editing and SSP. *Proceedings of the 13th International Modelica Conference, Regensburg, Germany, March 4–6, 2019*, 157, 69–78. <https://doi.org/10.3384/ecp1915769>
- OMRON. (2012). *PROFINET: Ethernet industrial abierto y estándar para la automatización*. 8.
- OpenModelica.org. (2019a). *OMSimulator's documentation*. <https://openmodelica.org/doc/OMSimulator/master/html/>
- OpenModelica.org. (2019b). *OMSimulator - FMI and TLM-Based Simulation and Co-simulation of External Models Warning*: <https://openmodelica.org/doc/OMSimulator/master/html/>
- Orchideo. (2019). *easySSP*. <https://www.easy-ssp.com/>
- OSP. (2018). *Open Simulation Platform – Joint Industry Project for the maritime industry*. <https://opensimulationplatform.com/>
- Osterrieder, P., Budde, L., & Friedli, T. (2020). The smart factory as a key construct of industry 4.0: A systematic literature review. *International Journal of Production Economics*, 221(November 2017), 107476. <https://doi.org/10.1016/j.ijpe.2019.08.011>
- Oyedeko, K. F. K., & Balogun, H. A. (2015). Modeling and Simulation of a Leak Detection for Oil and Gas Pipelines via Transient Model: A Case Study of the Niger Delta. *Journal of Energy Technologies and Policy*, 5(1), 16–27. <https://iiste.org/Journals/index.php/JETP/article/view/19389>
- Pang, T. Y., Restrepo, J. D. P., Cheng, C., Yasin, A., Lim, H., & Miletic, M. (2021). Developing a Digital Twin and Digital Thread Framework for an 'Industry 4.0' Shipyard. *Applied Sciences MDPI*.
- Patel, C., & Doshi, N. (2020). "a Novel MQTT Security framework in Generic IoT Model." *Procedia Computer Science*, 171(2019), 1399–1408. <https://doi.org/10.1016/j.procs.2020.04.150>
- Perabo, F., Park, D., Zadeh, M. K., Smogeli, O., & Jamt, L. (2020). Digital Twin Modelling of Ship Power and Propulsion Systems: Application of the

- Open Simulation Platform (OSP). *IEEE International Symposium on Industrial Electronics*, 2020-June, 1265–1270. <https://doi.org/10.1109/ISIE45063.2020.9152218>
- Perera, D. W. U., Winkler, D., & Skeie, N. O. (2016). Multi-floor building heating models in MATLAB and Modelica environments. *Applied Energy*, 171, 46–57. <https://doi.org/10.1016/j.apenergy.2016.02.143>
- Pérez-Cuadrado, E. G. (2016). Plan Made in China 2025. *Report ICEX España Exportación e Inversiones*.
- Pérez-Pérez, E. J., López-Estrada, F. R., Valencia-Palomo, G., Torres, L., Puig, V., & Mina-Antonio, J. D. (2021). Leak diagnosis in pipelines using a combined artificial neural network approach. *Control Engineering Practice*, 107(November 2020), 104677. <https://doi.org/10.1016/j.conengprac.2020.104677>
- PTC. (2019a). *Data Storage Options in Thingworx*. [https://doi.org/10.1007/978-1-4302-0682-8\\_6](https://doi.org/10.1007/978-1-4302-0682-8_6)
- PTC. (2019b). *ThingWorx Model Definition: Streams*. [https://support.ptc.com/help/thingworx\\_hc/thingworx\\_8\\_hc/en/index.html#page/ThingWorx/Help/Composer/DataStorage/Streams.html](https://support.ptc.com/help/thingworx_hc/thingworx_8_hc/en/index.html#page/ThingWorx/Help/Composer/DataStorage/Streams.html)
- PTC. (2019c). *ThingWorx Tutorials: Introduction to DataTables*. <https://community.ptc.com/t5/IoT-Tech-Tips/ThingWorx-Tutorials-Introduction-to-DataTables/td-p/532490>
- Pulido, B., Zamarreño, J. M., Merino, A., & Bregon, A. (2019). State space neural networks and model-decomposition methods for fault diagnosis of complex industrial systems. *Engineering Applications of Artificial Intelligence*, 79(November 2018), 67–86. <https://doi.org/10.1016/j.engappai.2018.12.007>
- Rajpurohit, L., & Verma, A. K. (2016). Industrie 4.0: An Overview. *International Journal of Advance Engineering and Research Development*, 3(3), 535–541. <https://doi.org/2348-4470>
- RCN. (2018). *A Strategy to Enable Canada 's Naval Team for the Digital Age*.
- Recamán Rivas, Á. (2018). Navantia's Shipyard 4.0 model overview. *Ciencia y Tecnología de Buques*, 11(22), 77. <https://doi.org/10.25043/19098642.165>

- Red Repply. (2020). *How To Take Advantage of Cloud Computing for Industrie 4.0*.
- Reis, J. Z., & Gonçalves, R. F. (2018). The role of internet of services (IoS) on industry 4.0 through the service oriented architecture (SOA). *IFIP Advances in Information and Communication Technology*, 536, 20–26. [https://doi.org/10.1007/978-3-319-99707-0\\_3](https://doi.org/10.1007/978-3-319-99707-0_3)
- Rifkin, J. (2011). La tercera revolución industrial. In *Paidós Ibérica*.
- Rodič, B. (2017). Industry 4.0 and the New Simulation Modelling Paradigm. *Organizacija*, 50(3), 193–207. <https://doi.org/10.1515/orga-2017-0017>
- Rossmann, M., Khadikar, A., Le Franc, P., Perea, L., Schneider-Maul, R., Buvat, J., & Ghosh, A. (2017). *Smart Factories : How can manufacturers realize the potential of digital industrial revolution*. 1–32. <https://doi.org/10.1097/NCQ.0000000000000292>
- Rubio, J. E., Roman, R., & Lopez, J. (2018). Analysis of cybersecurity threats in industry 4.0: The case of intrusion detection. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Vol. 10707 LNCS*. [https://doi.org/10.1007/978-3-319-99843-5\\_11](https://doi.org/10.1007/978-3-319-99843-5_11)
- Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088), 533–536. <https://doi.org/10.1038/323533a0>
- Samuel, A. L. (1960). Programming Computers to Play Games. *Advances in Computers*, 1(C), 165–192. [https://doi.org/10.1016/S0065-2458\(08\)60608-7](https://doi.org/10.1016/S0065-2458(08)60608-7)
- Sapna, S. (2012). *Backpropagation Learning Algorithm Based on Levenberg Marquardt Algorithm*. 393–398. <https://doi.org/10.5121/csit.2012.2438>
- Schirmann, M., Collette, M., & Gose, J. (2019). Ship motion and fatigue damage estimation via a digital twin. *Proceedings of the 6th International Symposium on Life-Cycle Civil Engineering, IALCCE 2018*, 2075–2082.
- Schweiger, G., Gomes, C., Engel, G., Hafner, I., Schoeggl, J., Posch, A., & Nouidui, T. (2019). An empirical survey on co-simulation: Promising standards, challenges and research needs. *Simulation Modelling Practice and Theory*, 95(i), 148–163. <https://doi.org/10.1016/j.simpat.2019.05.001>
- Sell, R., Rassölkin, A., Wang, R., & Otto, T. (2019). Integration of autonomous

- vehicles and industry 4.0. *Proceedings of the Estonian Academy of Sciences*, 68(4), 389–394. <https://doi.org/10.3176/proc.2019.4.07>
- Semeraro, C., Lezoche, M., Panetto, H., & Dassisti, M. (2021). Digital twin paradigm: A systematic literature review. *Computers in Industry*, 130. <https://doi.org/10.1016/j.compind.2021.103469>
- Shannon, R., & Johannes, J. D. (1976). Systems simulation: the art and science. *IEEE Transactions on Systems, Man and Cybernetics.*, 6(10), 723–724.
- Siemens-AMESIM. (2016). *Statechart Designer Combine discrete-event control and continuous system model*.
- Siemens-AMESIM. (2020a). *Simcenter Amesim Software: Hydraulic Resistance*.
- Siemens-AMESIM. (2020b). *Simcenter Amesim Software: Signal Control*.
- Siemens-AMESIM. (2020c). *Simcenter Amesim Software: Thermal-Hydraulic*.
- Siemens. (n.d.). *Gemelo digital*. Retrieved August 25, 2021, from <https://www.plm.automation.siemens.com/global/es/our-story/glossary/digital-twin/24465>
- Siemens. (2012). *PROFINET Descripción del sistema* (pp. 1–314).
- Siemens. (2018). MindSphere: Enabling the world's industries to drive their digital transformations. <https://www.plm.automation.siemens.com/global/en/products/mindsphere/internet-of-things-iiot.html>, 24. [www.siemens.com/mindsphere](http://www.siemens.com/mindsphere)
- Siemens. (2020a). *Future of Automation: Siemens Industrial Edge* (Issue February).
- Siemens. (2020b). *LiveTwin Edge app*.
- Siemens. (2020c). *MindSphere: The industrial IoT as a service solution*.
- Singh, H. (2020). Big data, industry 4.0 and cyber-physical systems integration: A smart industry context. *Materials Today: Proceedings*, xxxx. <https://doi.org/10.1016/j.matpr.2020.07.170>
- Sisinni, E., Saifullah, A., Han, S., Jennehag, U., & Gidlund, M. (2018). Industrial internet of things: Challenges, opportunities, and directions. *IEEE Transactions on Industrial Informatics*, 14(11), 4724–4734. <https://doi.org/10.1109/TII.2018.2852491>

- Slot, M., Huisman, P., & Lutters, E. (2020). A structured approach for the instantiation of digital twins. *Procedia CIRP*, 91, 540–545. <https://doi.org/10.1016/j.procir.2020.02.211>
- Smogeli, Ø. (2017). Digital twins at work in maritime and energy. *DNV-GL Feature, February*, 1–7.
- Soley, R. M. (2015). *Overview of the Industrial Internet Consortium*. June.
- Sousa, J., Ribeiro, L., Muranho, J., & Sá Marques, A. (2015). Locating leaks in water distribution networks with simulated annealing and graph theory. *Procedia Engineering*, 119(1), 63–71. <https://doi.org/10.1016/j.proeng.2015.08.854>
- Spragg, A., Mylon, P., Dolet, C., & Gottlieb, C. (2017). *Maritime in the new*. 12. <https://www.accenture.com/nl-en/insight-maritime-new>
- Stanic, V., Hadjina, M., Fafandjel, N., & Matulja, T. (2018). *Toward Shipbuilding 4.0 - An Industry 4.0 changing the face of the Shipbuilding Industry*. 69(3), 111–128. <https://doi.org/http://dx.doi.org/10.21278/brod69307> ISSN
- Sukarno, P., Adji Sidarto, K., Trisnobudi, A., Setyoadi, D. I., Rohani, N., & Darmadi, D. (2007). Leak Detection Modeling and Simulation for Oil Pipeline with Artificial Intelligence Method. *ITB Journal of Engineering Science*, 39(1), 1–19. <https://doi.org/10.5614/itbj.eng.sci.2007.39.1.1>
- Suresh, P., Vijay Daniel, J., Dr.V.Parthasarathy, & Aswathy, R. H. (2014). A state of the art review on the Internet of Things (IoT). History, Technology and fields of Deployment. *International Conference on Science, Engineering and Management Research (ICSEMR 2014)*.
- Thomas, G. (2008). Introduction to Modbus Serial and Modbus TCP. In *Contemporary Controls* (Vol. 9, Issue 5).
- Tiainen, T., Miettinen, J., Viitala, R., Hiekkänen, K., & Kuosmanen, P. (2019). Digital twin and virtual sensor for a rotor system. *Annals of DAAAM and Proceedings of the International DAAAM Symposium*, 30(1), 1115–1121. <https://doi.org/10.2507/30th.daaam.proceedings.156>
- Tiwana, M. (2019). *How to Quickly Create a Simulation-Based Digital Twin of an IIoT-Connected Product*. <https://www.ansys.com/blog/how-to-create-simulation-based-digital-twin-iiot-connected-product>
- Trauer, J., Schweigert-Recksiek, S., Engel, C., Spreitzer, K., & Zimmermann, M.

- (2020). WHAT IS A DIGITAL TWIN? - DEFINITIONS and INSIGHTS from AN INDUSTRIAL CASE STUDY in TECHNICAL PRODUCT DEVELOPMENT. *Proceedings of the Design Society: DESIGN Conference, 1*, 757–766. <https://doi.org/10.1017/dsd.2020.15>
- Trueba-Martinez, P., Ferrero-Gonzalez, S., Munín-Doce, A., Garcia, C., & Parente, S. (2020). Análisis de la implementación de un gemelo digital a bordo de un buque. *Ingeniería Naval*, 988.
- UK Trade & Investment. (2017). *MADE IN CHINA 2025 - China Manufacturing in the 21st Century - Opportunities for UK-China Partnership*.
- United Kingdom. Ministry of Defence. (2017). *National Shipbuilding Strategy: The Future of Naval Shipbuilding in the UK*. 56. [https://www.gov.uk/government/uploads/system/uploads/attachment\\_data/file/643873/NationalShipbuildingStrategy\\_lowres.pdf](https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/643873/NationalShipbuildingStrategy_lowres.pdf)
- Forecasting trends & challenges for a 4.0 Shipbuilding Workforce in Europe, (2020).
- USWE. (2020). *Results of USWE 's Survey on Skills , Education and 4 . 0 technologies in Shipbuilding*.
- Value, I., & Initiative, C. (2019). *Connected Industries Open Framework for Industrial Value Chain Transformation*. April, 1–5.
- VanDerHorn, E., & Mahadevan, S. (2021a). Digital Twin: Generalization, characterization and implementation. *Decision Support Systems*, 145(February), 113524. <https://doi.org/10.1016/j.dss.2021.113524>
- VanDerHorn, E., & Mahadevan, S. (2021b). *Digital Twin Definitions in the Literature (Attached to: Digital Twin: Generalization, characterization and implementation)*.
- Vega, M. C., Pablo Oliete Vivas, Christian Morales Rios, Carlos González Luis, Bruno Cendón Martín, & Alberto Hernández Seco. (2015). *Las Tecnologías IoT dentro de la Industria Conectada 4.0*. 190. <https://www.eoi.es/es/savia/publicaciones/21125/las-tecnologias-iot-dentro-de-la-industria-conectada-40>
- Verde, C., Molina, L., & Torres, L. (2014). Parameterized transient model of a pipeline for multiple leaks location. *Journal of Loss Prevention in the Process Industries*, 29(1), 177–185. <https://doi.org/10.1016/j.jlp.2014.02.013>



- Voith, A. (2018). *Royal Australian navy awards major bridge simulator contract to Kongsberg Digital*.
- Vöth, S., & Vasilyeva, M. (2020). Potential of Modelica for the creation of digital twins. *Advances in Raw Material Industries for Sustainable Development Goals*, 386–389. <https://doi.org/10.1201/9781003164395-48>
- Wang, P., & Luo, M. (2021). A digital twin-based big data virtual and real fusion learning reference framework supported by industrial internet towards smart manufacturing. *Journal of Manufacturing Systems*, 58(PA), 16–32. <https://doi.org/10.1016/j.jmsy.2020.11.012>
- Wang, Q., Zhu, X., Ni, Y., Gu, L., & Zhu, H. (2020). Blockchain for the IoT and industrial IoT: A review. *Internet of Things*, 10(66), 100081. <https://doi.org/10.1016/j.iot.2019.100081>
- Wang, Y., Steinbach, T., Klein, J., & Anderl, R. (2021). Integration of model based system engineering into the digital twin concept. *Procedia CIRP*, 100, 19–24. <https://doi.org/10.1016/j.procir.2021.05.003>
- Waslo, R., Lewis, T., Hajj, R., & Carton, R. (2017). Industry 4.0 and cybersecurity. Managing risk in an age of connected production. In *Deloitte University Press* (Vol. 1). <https://www2.deloitte.com/us/en/insights/focus/industry-4-0/cybersecurity-managing-risk-in-age-of-connected-production.html><https://dupress.deloitte.com/dup-us-en/focus/industry-4-0/cybersecurity-managing-risk-in-age-of-connected-production.html>
- Wetter, M. (2018). *Overview of Modelica Buildings Library*.
- Wetter, M., Zuo, W., & Nouidui, T. S. (2011). Recent Developments of the Modelica “Buildings” Library for Building Energy and Control Systems. *Proceedings from the 8th International Modelica Conference, Technical University, Dresden, Germany*, 63, 266–275. <https://doi.org/10.3384/ecp11063266>
- Wiktorsson, M., Noh, S. Do, Bellgran, M., & Hanson, L. (2018). Smart Factories: South Korean and Swedish examples on manufacturing settings. *Procedia Manufacturing*, 25, 471–478. <https://doi.org/10.1016/j.promfg.2018.06.128>
- Workshop, I. I., & Angeles, L. (2015). *Fmi Workshop*. 1–40.
- World Economic Forum. (2018). The next economic growth engine: Scaling

- Fourth Industrial Revolution technologies in production. *World Economic Forum, January*, 33. [https://www.weforum.org/whitepapers/the-next-economic-growth-engine-scaling-fourth-industrial-revolution-technologies-in-production%0Ahttp://www3.weforum.org/docs/WEF\\_Technology\\_and\\_Innovation\\_The\\_Next\\_Economic\\_Growth\\_Engine.pdf](https://www.weforum.org/whitepapers/the-next-economic-growth-engine-scaling-fourth-industrial-revolution-technologies-in-production%0Ahttp://www3.weforum.org/docs/WEF_Technology_and_Innovation_The_Next_Economic_Growth_Engine.pdf)
- Wortmann, A., Combemale, B., & Barais, O. (2017). A Systematic Mapping Study on Modeling for Industry 4.0. *Proceedings - ACM/IEEE 20th International Conference on Model Driven Engineering Languages and Systems, MODELS 2017*, 281–291. <https://doi.org/10.1109/MODELS.2017.14>
- WP6-OECD. (2018). Peer Review of the Finnish Shipbuilding Industry. *Oecd 2018*.
- Wübbecke, J., Meissner, M., Zenglein, M. J., Ives, J., & Conrad, B. (2016). Made in China 2025 The making of a high-tech superpower and its implications for industrial countries. *MERICs Papers on China, 2*, 76. [https://www.merics.org/fileadmin/user\\_upload/downloads/MPOC/MP-OC\\_Made\\_in\\_China\\_2025/MPOC\\_No.2\\_MadeinChina\\_2025.pdf](https://www.merics.org/fileadmin/user_upload/downloads/MPOC/MP-OC_Made_in_China_2025/MPOC_No.2_MadeinChina_2025.pdf)
- Zadkarami, M., Shahbazian, M., & Salahshoor, K. (2016). Pipeline leakage detection and isolation: An integrated approach of statistical and wavelet feature extraction with multi-layer perceptron neural network (MLPNN). *Journal of Loss Prevention in the Process Industries, 43*, 479–487. <https://doi.org/10.1016/j.jlp.2016.06.018>
- Zadkarami, M., Shahbazian, M., & Salahshoor, K. (2017). Pipeline leak diagnosis based on wavelet and statistical features using Dempster–Shafer classifier fusion technique. *Process Safety and Environmental Protection, 105*, 156–163. <https://doi.org/10.1016/j.psep.2016.11.002>
- Zhang, J. (1997). Designing a cost-effective and reliable pipeline leak-detection system. *Pipes and Pipelines International, 42*(1), 20–26.
- Zhang, Q., Wu, F., Yang, Z., Li, G., & Zuo, J. (2019). Simulation of the transient characteristics of water pipeline leakage with different bending angles. *Water (Switzerland), 11*(9). <https://doi.org/10.3390/w11091871>
- Zhao, X., Ba, Q., Zhou, L., Li, W., & Ou, J. (2014). BP neural network recognition algorithm for scour monitoring of subsea pipelines based on active thermometry. *Optik, 125*(18), 5426–5431.

<https://doi.org/10.1016/j.ijleo.2014.06.018>

## 8. APÉNDICE A

Compo- nente	Variable	Descripción	Modelo de comportamiento
	CIStemp	Temperatura del espacio exterior a la cabina.	
	EXtemp	Temperatura exterior	
	EXfan	Condición de encendido o apagado de un ventilador para renovación del aire en el interior de la cabina	
<b>Ambiente</b>	AMBT	Temperatura de la cabina	
<b>(cabina)</b>	AMBH	Humedad de la cabina	
<b>Bomba Centrífuga</b>	CPAH	Horas acumuladas de trabajo de la bomba	
	CPCF	Caudal de agua calculado de la bomba	<i>Curva de la bomba. f(CPPR)</i>
	CPDP	Presión de descarga de la bomba	
	CPHE	Eficiencia hidráulica de la bomba	<i>CPHP/CPMMP</i>
	CPHP	Potencia hidráulica de la bomba	<i>CPPR x MVWF x 27,8</i>
	CPMC	Intensidad eléctrica del motor de la bomba	
	CPMEP	Potencia eléctrica del motor de la bomba	<i>Voltaje x Intensidad x 1,7 x 0,9</i>
	CPMMP	Potencia mecánica del motor de la bomba	<i>CPMEP / 0,91</i>
	CPMT	Temperatura de la bomba centrífuga	
	CPMV	Voltaje del motor de la bomba	
	CPPR	Incremento de presión en la bomba	<i>CPDP - CPSP</i>
CPSP	Presión de aspiración de la bomba		

	CPST	Estado de la bomba (ON/OFF)	
<b>Tanque</b>	HTLS	Nivel de llenado del tanque	
<b>Moto-vál- vula</b>	MVCV	Coefficiente Kv de la válvula	<i>f(MVSO). Tabla de la válvula</i>
	MVDP	Presión diferencial en la válvula	
	MVOT	Par de la válvula	
	MVSO	Apertura de la válvula	
	MVWF	Caudal de agua en la válvula	<i>f(MVSO). Tabla de la válvula</i>
<b>Arrancador</b>	STT	Temperatura del cuadro de control	
<b>Unit-cooler</b>	UCAF	Caudal de aire en el unit-cooler	
	UCAH	Horas de trabajo acumuladas en el unit-cooler	
	UCAIT	Temperatura de entrada de aire en el unit-cooler	
	UCAOT	Temperatura de salida de aire en el unit-cooler	
	UCFMC	Intensidad eléctrica en el motor del ventilador	
	UCFMP	Potencia mecánica en el motor del ventilador	
	UCFMS	Velocidad de giro del motor del ventilador	
	UCFMT	Temperatura del motor del ventilador	
	UCFMV	Voltaje del motor del ventilador	
	UCFS	Velocidad de giro del ventilador	
	UCFVB	Vibraciones del ventilador	
	UCHC	Intensidad eléctrica del calentador	
	UCHS	Estado del calentador (ON/OFF)	
	UCHV	Voltaje del calentador	
	UCMVB	Vibraciones del motor del ventilador	
UCSDP	Air cooler air filter differential pressure sensor		

UCWDP	Air cooler water differential pressure sensor	
UCWF	Caudal de agua en el unit-cooler	$(UCWDP/Kv)^{1/2}$
UCWIT	Temperatura de entrada de agua en el unit-cooler	
UCWOT	Temperatura de salida de agua en el unit-cooler	
UCWRF	Caudal de agua requerido en el unit-cooler	

## 9. LISTADO DE FIGURAS

---

Imagen 2-1: Tecnologías Habilitadoras (KETs).....	26
Imagen 2-2: Tecnologías clave en el desarrollo del gemelo digital .....	30
Imagen 2-3: Representación de Michael Grieves sobre el gemelo digital y el flujo de datos e información entre el activo físico y el virtual .....	36
Imagen 3-1: Definición de gemelo digital .....	47
Imagen 3-2: Tecnologías habilitadoras y funcionalidades principales de un gemelo digital.....	49
Imagen 3-3: Niveles de gemelo digital según Ashtari .....	50
Imagen 3-4: Niveles (según grado de madurez) en el gemelo digital. (Madni et al., 2019) .....	51
Imagen 3-5: Esquema Ciclo de vida de un producto vs etapas creación gemelo digital.....	53
Imagen 3-6: Componentes principales para la implantación de un gemelo digital.....	60
Imagen 3-7: Componentes de capa física e interfaz hacia el gemelo digital .....	63
Imagen 3-8: Componentes de un gemelo digital (Mundo virtual).....	66
Imagen 3-9: Planteamiento esquemático para gemelo digital del buque desde el punto de vista del buque como agregado de sistemas.....	75
Imagen 3-10: Ejemplo de gemelo digital para varios sistemas del buque, con diferentes funcionalidades.....	78
Imagen 3-11: Diseño para interfaz gemelo digital del buque - pantalla principal.....	80
Imagen 3-12: Diseño para interfaz gemelo digital del buque – pantalla gemelo digital de sistema .....	81
Imagen 4-1: Esquema del sistema HVAC .....	91
Imagen 4-2: 1ª arquitectura planteada: gemelo digital centralizado en plataforma de IoT .....	94

Imagen 4-3: 2ª arquitectura planteada: gemelo digital descentralizado ..	95
Imagen 4-4: Arquitectura despliegue gemelo digital (gemelo digital centralizado en plataforma IoT).....	96
Imagen 4-5: Componentes/funcionalidades planteados en el gemelo digital .....	97
Imagen 4-6: Diagrama de variables para el modelo de datos (SysML)....	98
Imagen 4-7: Diagrama de secuencias para las peticiones desde el gemelo digital al sistema de control del buque .....	102
Imagen 4-8: Esquema del modelo desarrollado para modo invierno ....	105
Imagen 4-9: Evolución de la temperatura de la habitación (AMBT) de acuerdo con el modelo para calefacción desarrollado en OpenModelica.....	106
Imagen 4-10: Esquema del modelo desarrollado para modo verano .....	107
Imagen 4-11: Evolución de la temperatura de la habitación (AMBT) de acuerdo con el modelo para modo verano desarrollado en OpenModelica	109
Imagen 4-12: Interfaz para configurar la exportación a FMU desde OpenModelica.....	110
Imagen 4-13: Componentes internos de un paquete FMU.....	111
Imagen 4-14: Diagrama de secuencia para ejecutar simulaciones desde el gemelo digital .....	112
Imagen 4-15: Diagrama de actividades: Ejecución de FMU en servidor de simulaciones, librería PyFMI .....	113
Imagen 4-16: Conexiones entre los widgets de las interfaces y los servicios .....	117
Imagen 4-17: Arquitectura descentralizada.....	120
Imagen 4-18: Esquema del modelo de simulación multifísico .....	122
Imagen 4-19: Componente para modelar una bomba centrífuga en AMESIM .....	123
Imagen 4-20: Componente para modelar una válvula en AMESIM.....	124



Imagen 4-21: Componentes para modelar un intercambiador de calor en AMESIM .....	125
Imagen 4-22: Interfaz de la herramienta de regresión .....	126
Imagen 4-23: Componentes para simular un flujo de calor y un volumen de control (cámara de aire).....	127
Imagen 4-24: Componente para simular una fuente de aire .....	128
Imagen 4-25: Componente para simular una cámara de aire con un puerto térmico que admite múltiples entradas.....	128
Imagen 4-26: Componentes empleados para simular las paredes de la habitación y la transferencia de calor con el exterior .....	129
Imagen 4-27: Componentes empleados para simular la transferencia de calor desde el cuadro eléctrico y el calor generado por los motores de la bomba y del ventilador .....	129
Imagen 4-28: Diagrama de estados definido en AMESIM para simular el control ON/OFF del sistema .....	130
Imagen 4-29: Variables y eventos definidos en el diagrama de estados	131
Imagen 4-30: Esquema resumido del diagrama de estados para calcular la velocidad del ventilador con las variables de entrada y salida. ....	132
Imagen 4-31: Representación en AMESIM de los componentes creados para definir el control de la válvula.....	132
Imagen 4-32: Comparativa de los resultados obtenidos para la evolución de la temperatura en la cabina para el modo invierno .....	133
Imagen 4-33: Comparativa de los resultados obtenidos para la evolución de la temperatura en la cabina para el modo verano .....	134
Imagen 4-34: Diagrama de bloques (SysML) para definición de entradas y salidas de los modelos de simulación.....	135
Imagen 4-35: Herramienta “Performance Analyzer” .....	137
Imagen 4-36: Comparativa entre los valores real y simulado para la variable CPPR (incremento de presión en la bomba) .....	138

Imagen 4-37: Comparativa entre los valores real y simulado para la variable MVWF (caudal de agua en el circuito hidráulico) .....	139
Imagen 4-38: Comparativa entre los valores real y simulado para la variable MVDP (caída de presión en la válvula) .....	139
Imagen 4-39: Comparativa entre los valores real y simulado para la variable UCWDP (caída de presión en el serpentín del unit-cooler).....	139
Imagen 4-40: Bloque definido en AMESIM para poder realizar la exportación de un modelo de simulación a formato FMU (modelo para predicción).....	140
Imagen 4-41: Bloque definido en AMESIM para poder realizar la exportación de un modelo de simulación a formato FMU (modelo para tiempo real).....	143
Imagen 4-42: Visualización de interfaz Industrial Edge de Siemens: LiveTwin - Información de modelo FMU.....	145
Imagen 4-43: Ventana de Industrial Edge de Siemens: Flowcreator .....	146
Imagen 4-44: Visualización de señales reales frente a señales simuladas, indicadores de error entre valores reales y simulados y ventana de alertas (sobre WinCC OA) .....	147
Imagen 4-45: Pruebas de detección de fallos (1/3).....	148
Imagen 4-46: Pruebas de detección de fallos (2/3).....	149
Imagen 4-47: Pruebas de detección de fallos (3/3).....	150
Imagen 4-48: Co-simulación de FMU en SIMULINK .....	154
Imagen 4-4-49: Co-simulación de FMU en AMESIM.....	154
Imagen 4-50: Co-simulación de FMU en Edge LiveTwin.....	154
Imagen 4-51: FMUs desarrollados para emplear en las pruebas con SSP y OSP .....	157
Imagen 4-52: Orquestación de FMU en OMSimulator .....	158
Imagen 4-53: Gráfica de salida de resultados de co-simulación en la plataforma OMSimulator .....	159

Imagen 4-54: Esquema de la conexión de FMUs para simulación en el entorno kopl.....	161
Imagen 4-55: Visualización de los resultados obtenidos para la variable UCAOT en el entorno Kopl.....	162
Imagen 4-56: Visualización de los resultados obtenidos para la variable MVWF en el entorno Kopl.....	162
Imagen 5-1: Esquema general de una neurona.....	171
Imagen 5-2: Esquema general de una red neuronal.....	172
Imagen 5-3: Propagación del error en el algoritmo de backpropagation.....	175
Imagen 5-4: Esquema simplificado del sistema contraincendios.....	179
Imagen 5-5: Ejemplo de ramales del modelo de simulación del sistema.....	180
Imagen 5-6: Modo de fallo. Identificación tubería dañada + sección inoperativa.....	181
Imagen 5-7: Resultados de la simulación (caída de presión) realizada para la apertura del cañón contraincendios.....	181
Imagen 5-8: Resultados de la simulación (caída de presión) realizada para la apertura del eyector de achique N°2.....	182
Imagen 5-9: Instalación provisional para pruebas.....	184
Imagen 5-10: Señales de presión recogidas en los sensores del sistema contraincendios por agua salada del buque (prueba 1).....	186
Imagen 5-11: Señales de presión recogidas en los sensores del sistema contraincendios por agua salada del buque (prueba 1). Vista de detalle.....	186
Imagen 5-12: Señales de presión recogidas en los sensores del sistema contraincendios por agua salada del buque (prueba 4).....	187
Imagen 5-13: Comparativa resultados (sensor PT007) real vs simulados (prueba1).....	189
Imagen 5-14: Comparativa resultados (sensor PT007) real vs simulados (prueba3).....	189

Imagen 5-15: Comparativa de los resultados obtenidos para la caída de presión en un sensor tipo (real vs. simulado) .....	190
Imagen 5-16: Esquema general para una red neuronal .....	192
Imagen 5-17: Esquema general de la red neuronal empleada (número de capas y neuronas por capa por definir).....	193
Imagen 5-18: Diagrama de flujo del código desarrollado para entrenamiento de la red para detección de fallos.....	196
Imagen 5-19: Regresión obtenida con 1 capa y 20 neuronas.....	199
Imagen 5-20: Regresión para 1 capa oculta y 15 neuronas.....	201
Imagen 5-21: Caracterización apertura de eyectores.....	204
Imagen 5-22: Caracterización apertura rociadores.....	204
Imagen 5-23: “simulación de rotura” en boquil CI y apertura normal de un boquil contraincendios .....	205
Imagen 5-24: Apertura eyectores. Parada/arranque bomba CI .....	205
Imagen 5-25: Resultados obtenidos para 5 capas y 5 neuronas y para 7 capas con 5 neuronas (tamaño de muestra: 20 elementos) .....	213
Imagen 5-26: Resultados obtenidos para 4 capas y 5 neuronas y para 7 capas con 5 neuronas (tamaño de muestra: 35 elementos) .....	213
Imagen 5-27: Resultados obtenidos para 2 capas y 5 neuronas y para 8 capas con 5 neuronas (tamaño de muestra: 50 elementos) .....	213
Imagen 5-28: Propuesta de división en zonas del buque para etiquetado de localización de fallos.....	214
Imagen 5-29: Caídas de presión en funcionamiento normal .....	215
Imagen 5-30: Caídas de presión en rotura de tubería .....	215
Imagen 5-31: Diagrama de flujo del código desarrollado para entrenamiento de la red para localización de fallos.....	217
Imagen 5-32: Desarrollo de ventanas para interfaces (mashups) en Thingworx .....	223

Imagen 5-33: Pruebas de diseño de interfaces para monitorización del sistema (funcionamiento real) .....224

Imagen 5-34: Pruebas de diseño de interfaces para monitorización del sistema (fallo del sistema - alarma).....224

Imagen 6-1: Tecnologías habilitadoras y funcionalidades principales de un gemelo digital.....228

## 10. LISTADO DE TABLAS

---

Tabla 2-1: Evolución temporal de las diferentes tipologías de gemelos digitales. Adaptada de (Erikstad, 2018). .....	40
Tabla 3-1. Parámetros del gemelo digital (estimación) (Trueba-Martinez et al., 2020) .....	83
Tabla 3-2: Resumen de ubicación de componentes del gemelo digital ....	87
Tabla 4-1: Datos experimentales necesarios como entradas para la herramienta HHX Regression Tool .....	127
Tabla 4-2: Tabla en la que se definen las condiciones de velocidad del ventilador .....	132
Tabla 4-3: Casos de funcionamiento real del sistema empleados para la validación del modelo de simulación.....	138
Tabla 4-4: Herramientas que implementan el estándar SSP como complemento al FMI .....	157
Tabla 4-4-5: Comparativa de funcionalidades en las herramientas analizadas para co-simulación de modelos FMU .....	166
Tabla 5-1: Referencias recientes uso de redes neuronales para detección de fallos en tuberías. Basado en (Pérez-Pérez et al., 2021).....	170
Tabla 5-2: Listado de aperturas de consumidores para caracterización de comportamiento normal (simulación).....	191
Tabla 5-3: Ubicaciones de roturas de tuberías para caracterización de comportamiento en fallo (simulación) .....	191
Tabla 5-4: Resumen de resultados obtenidos en el entrenamiento de la red neuronal (selección de casos entrenamiento/validación/test manual) .....	198
Tabla 5-5: Resumen de resultados obtenidos en el entrenamiento de la red neuronal (selección de casos entrenamiento/validación/test aleatoria) .....	200

Tabla 5-6: Resultados obtenidos en el entrenamiento de la red neuronal (tamaño de la muestra de 3.5 s y para 1,2,3 capas).....	208
Tabla 5-7: Regresiones obtenidas en los entrenamientos de la red neuronal .....	210
Tabla 5-8: Caídas de presión relativas (en bares) en los boquiles CI N° 58 y N° 59 (a los 5 s de la apertura y a los 10 segundos de la apertura) .....	216
Tabla 5-9: Resumen de resultados obtenidos para el entrenamiento de la red neuronal empleando la función de entrenamiento: Scaled conjugate gradient .....	218
Tabla 5-10: Resumen de resultados obtenidos para el entrenamiento de la red neuronal empleando la función de entrenamiento: Levenberg-Marquard .....	219
Tabla 5-11: Resumen de resultados obtenidos para el entrenamiento de la red neuronal empleando la función de entrenamiento: Regularización Bayesiana .....	220