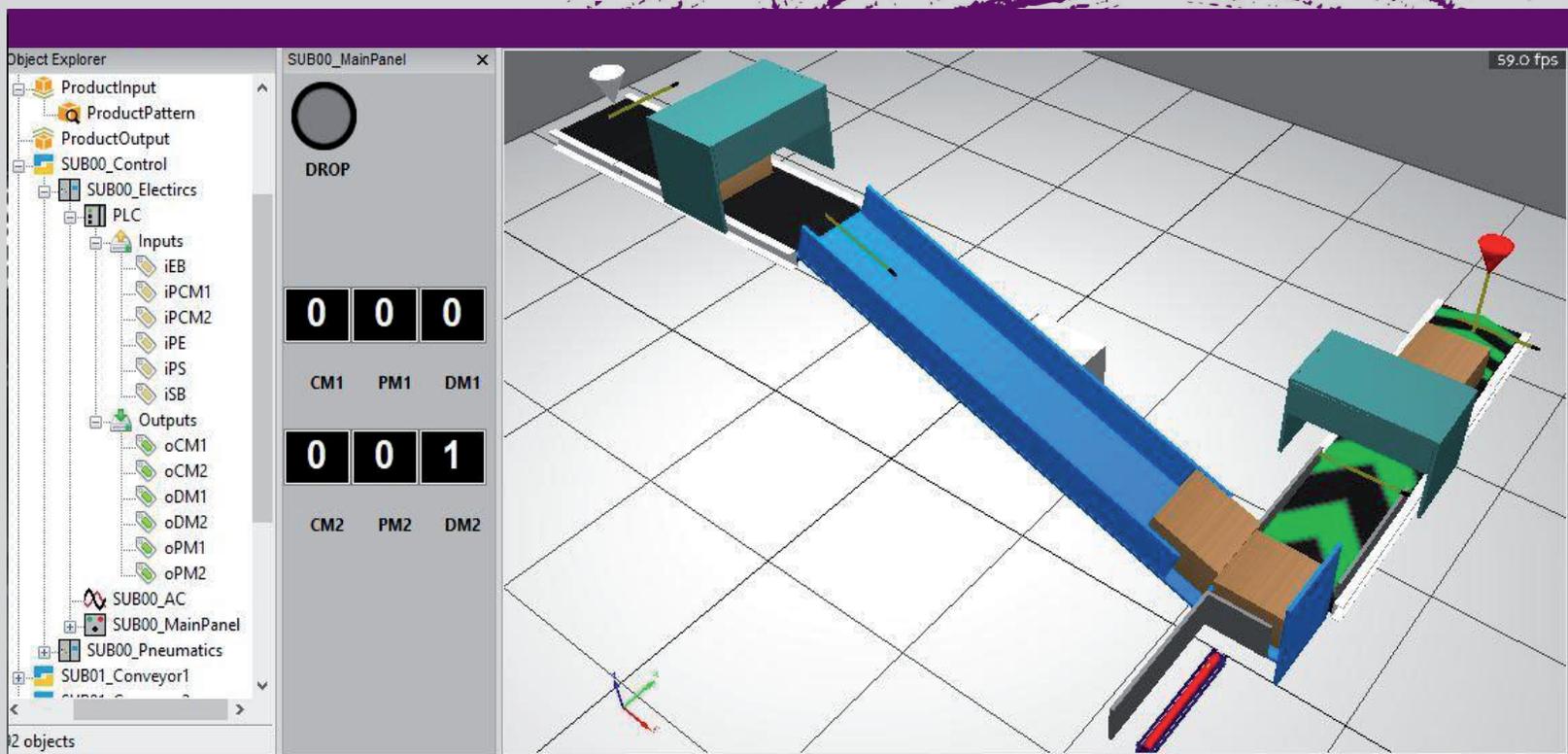




**Mondragon
Unibertsitatea**

TESIS DOCTORAL

APLICACIÓN DE TECNOLOGÍAS DE DIGITALIZACIÓN A SISTEMAS INDUSTRIALES AUTOMATIZADOS BASADOS EN CONTROLADORES LÓGICOS PROGRAMABLES



IGOR AZKARATE FERNANDEZ | Arrasate-Mondragón, 2023



Mondragon **Goi Eskola**
Unibertsitatea **Politeknikoa**

TESIS DOCTORAL

**Aplicación de tecnologías de digitalización
a sistemas industriales automatizados basados
en controladores lógicos programables**

Autor:

Igor AZKARATE FERNÁNDEZ

Director:

Dr. Luka ECIOLAZA ECHEVERRÍA

*Tesis presentada para la obtención del título de
Doctor en Ingeniería Electrónica*

en el

Grupo de Investigación en Robótica y Automatización
Departamento de Electrónica e Informática

Escuela Politécnica Superior
Mondragon Unibertsitatea

11 de noviembre de 2022



Mondragon **Goi Eskola**
Unibertsitatea **Politeknikoa**

TESIS DOCTORAL

**Aplicación de tecnologías de digitalización
a sistemas industriales automatizados basados
en controladores lógicos programables**

Autor:

Igor AZKARATE FERNÁNDEZ

Director:

Dr. Luka ECIOLAZA ECHEVERRÍA

*Tesis presentada para la obtención del título de
Doctor en Ingeniería Electrónica*

en el

Grupo de Investigación en Robótica y Automatización
Departamento de Electrónica e Informática

Escuela Politécnica Superior
Mondragon Unibertsitatea

11 de noviembre de 2022

A Eneritz, Miren y Paul.

Declaración de originalidad

Yo, Igor AZKARATE FERNÁNDEZ, declaro que esta tesis es original, fruto de mi trabajo personal, y que no ha sido previamente presentada para obtener otro título o calificación profesional. Las ideas, formulaciones, imágenes, ilustraciones tomadas de fuentes ajenas han sido debidamente citadas y referenciadas.

En Arrasate-Mondragón, a 11 de noviembre de 2022.

Agradecimientos

Aunque se necesitarían muchas páginas para agradecer a todas las personas con una influencia clave en este proceso, me gustaría realizar las siguientes menciones:

- A Luka Eciolaza, por su apoyo, positivismo, confianza, visión y disposición en cualquier momento y fase de este trabajo.
- A Nestor Arana, por su contribución en la propuesta de tesis y su puesta en marcha, y a Aitor Aguirre, Mikel Ayani, Juan Carlos Mugarza y Josu Uranga, por sus aportaciones a la investigación y a los artículos publicados.
- A Josu Galarza, Roman Viteri y Joseba Zaldibia, por haber apostado por mí allá por 2008, abriéndome las puertas de la Universidad y de un mundo apasionante, así como por toda la ayuda prestada desde entonces.
- A la Universidad, por la oportunidad que se me ha brindado para el desarrollo de este trabajo, así como por los medios puestos a mi disposición. Y a todos aquellos compañeros con quienes he compartido todos esos pequeños momentos en el trabajo diario, y que tanto oxígeno me dan.
- A todas y cada una de las personas que han integrado o integran nuestro equipo de trabajo, tanto área como grupo de investigación.
- A Javier Oyarzun, Dani Reguera e Iñaki Vélez de Mendizabal, por las inquietudes compartidas y sus consejos durante el desarrollo de nuestras respectivas tesis doctorales.
- Al personal sanitario que me atendió durante el primer año de este trabajo, como consecuencia de un problema grave de salud, en el Hospital de Mendaro en primera instancia, y a domicilio después, y que hizo posible que lo retomara con más fuerza incluso que antes.
- A Eneritz, por haberme dado el empujón que necesitaba para embarcarme en esta aventura y por su apoyo; y a Miren y a Paul por su alegría y por lo rápido que me ayudan a desconectar. Espero hacerlos más caso por las noches de ahora en adelante.
- A mi familia, por su respaldo. A mi familia política, por su acogida. A mis amigos de siempre y a los que se han ido incorporando.
- A todos los que están, y a todos los que siguen estando, en especial a aitxitxa Pepe, a mi amigo Haika y a mi suegro José Alberto. Beti gogoan.

Muchas gracias a todos/as. Bihotzez, eskerrik asko.

Resumen - Laburpena - Abstract

En un entorno industrial en el que los plazos de entrega de soluciones automatizadas suponen un diferencial en el mercado, una de las principales preocupaciones del personal técnico de empresas desarrolladoras es agilizar el desarrollo, testado y validación de código de dispositivos de control. La aplicación de tecnologías de digitalización supone un cambio de paradigma no sólo en el testado y validación de software, sino en el apoyo al diseño de nuevas metodologías que sistematicen la programación y reduzcan los errores en su origen. En este trabajo de tesis se experimenta con la emulación de sistemas industriales automatizados como herramienta para agilizar las fases mencionadas, así como en otras del ciclo de vida de dichos sistemas.

In an industrial environment in which delivery times of automated solutions are a differential in the market, one of the main concerns of the technical staff of development companies is to speed up the development, testing and validation of control device code. The application of digitization technologies represents a paradigm shift not only in the testing and validation of software, but also in supporting the design of new methodologies that systematize programming and reduce errors at their source. In this thesis work the emulation of automated industrial systems is experimented with as a tool to speed up the mentioned phases, as well as in others of the lifecycle of such systems.

Industri automatizazio arloko proiektuen entrega-epEEK merkatuan positzionatze on baten giltza duten testuinguruan, kontrolagailuen kodearen garapen, testatze eta balioztatzea arintzea da teknikarien buruhausteetako bat. Digitalizazio-teknologien aplikazioak paradigma aldaketa bat dakar ez bakarrik softwarearen testatze eta balioztatzean, baita programazioaren sistematizazio eta akatsen murrizketarako metodologia berrien diseinua lagunduz. Tesi lan honetan sistema industrial automatizatu-
tuen emulazioa probatzen da, bai aipatutako faseak bizkortzeko erreminta bezala, bai sistema hauen bizitza-erabilgarriko beste batzuetan ezarrita.

Tabla de contenidos

I Enmarque de la tesis	1
1. Introducción	3
1.1. Motivación y alcance de la investigación	3
1.2. Antecedentes	6
1.2.1. Sistemas industriales automatizados	6
1.2.2. Sistemas de control basados en PLC	7
1.2.3. Testado y validación convencional de sistemas de control basados en PLC	9
1.2.4. Las tecnologías de digitalización en el marco de la Industria 4.0	9
1.2.4.1. La cuarta revolución industrial	10
1.2.4.2. Conceptos de interés	11
1.2.5. El soporte del gemelo digital en sistemas industriales automatizados	13
1.2.5.1. Testado y validación de código de PLC en un gemelo digital	13
1.2.5.2. El gemelo digital más allá del testado y validación de código de PLC	14
1.2.6. Conectividad entre el dispositivo de control y el gemelo digital del sistema industrial automatizado	14
1.2.7. Introducción a algunas herramientas de emulación disponibles en el mercado	15
2. Estado del arte	19
2.1. Caracterización de la emulación de sistemas industriales automatizados	19
2.1.1. El concepto de gemelo digital	19
2.1.2. Conectividad entre el dispositivo de control y el gemelo digital	20
2.1.3. Técnicas de modelado	21
2.2. El reto de agilizar el desarrollo, el testado y la puesta en servicio de sistemas de control	21
2.2.1. Antecedentes: puesta en servicio convencional	21
2.2.2. La irrupción del gemelo digital: puesta en servicio virtual y otras aplicaciones	22
2.2.2.1. Puesta en servicio virtual basada en un gemelo digital	22
2.2.2.2. Otras aplicaciones del gemelo digital	23
2.2.3. Intentos de sistematización del desarrollo de código	25
2.3. Fortalezas y debilidades	26
2.4. Líneas de investigación identificadas	28
2.4.1. Implementación sistemática y sencilla de sistemas dinámicos de eventos discretos en PLCs	28

2.4.2.	El gemelo digital más allá de la validación de código de sistemas de control	29
2.4.3.	La emulación a nivel componente como soporte a la determinación de parámetros operativos en la fase de puesta en servicio virtual	29
2.4.4.	Otras oportunidades de investigación	30
2.5.	Definición del trabajo	31
3.	Objetivos, materiales y métodos utilizados	33
3.1.	Objetivos e hipótesis	33
3.1.1.	Definición de objetivos	33
3.1.1.1.	Objetivo principal	33
3.1.1.2.	Objetivos parciales	33
3.1.2.	Definición de hipótesis	35
3.2.	Materiales	35
3.3.	Metodología	36
II	Desarrollo de la tesis	39
4.	Introducción al desarrollo de la tesis	41
5.	Marco para testado y puesta en servicio virtual de programas de control de sistemas industriales automatizados.	45
5.1.	Introducción	46
5.1.1.	Automatización de sistemas industriales	46
5.1.1.1.	Testado y validación convencional de sistemas de control	46
5.1.1.2.	El gemelo digital en el testado y validación de sistemas de control	46
5.1.2.	Definición del problema	48
5.2.	Evaluación empírica	48
5.2.1.	Pregunta de investigación	48
5.2.2.	Caracterización del ensayo	48
5.2.2.1.	Material utilizado	49
5.2.2.2.	Caracterización del modelo virtual	49
5.2.2.3.	Trabajo realizado	52
5.2.2.4.	Metodología	53
5.2.2.5.	Cierre del experimento	54
5.2.3.	Resultados	54
5.2.4.	Discusión	56
5.3.	Trabajo relacionado	57
5.4.	Conclusión y trabajo futuro	58
6.	Implementación semi-compilada de redes de Petri en PLCs.	59
6.1.	Introducción	59
6.1.1.	Sistemas dinámicos de eventos discretos	59
6.1.2.	Modelado de sistemas dinámicos de eventos discretos	62
6.1.2.1.	Modelado basado en redes de Petri	62
6.1.2.2.	Implementación de redes de Petri en sistemas industriales automatizados	64

6.1.2.3.	Modelado basado en metodología GRAFCET	65
6.1.2.4.	Implementación de GRAFCET en sistemas industria- les automatizados	65
6.1.3.	Comparativa entre ambos enfoques	65
6.1.4.	Definición del problema	66
6.2.	Evaluación empírica	67
6.2.1.	Preguntas de investigación	68
6.2.2.	Caracterización del ensayo	68
6.2.2.1.	Material utilizado	68
6.2.2.2.	Caracterización del modelo virtual	69
6.2.2.3.	Trabajo realizado	71
6.2.2.4.	Metodología	74
6.2.2.5.	Cierre del experimento	79
6.2.3.	Resultados	79
6.2.4.	Discusión	81
6.3.	Trabajo relacionado	83
6.4.	Conclusión y trabajo futuro	83
7.	Diagnóstico de actividad	
	en sistemas industriales automatizados.	85
7.1.	Introducción	85
7.1.1.	Antecedentes y enmarque	86
7.1.2.	Fundamentos del marco	87
7.1.2.1.	Niveles lógicos	88
7.1.2.2.	Núcleo de ejecución de ULISES	89
7.1.3.	Definición del problema	90
7.2.	Evaluación empírica	90
7.2.1.	Preguntas de investigación	90
7.2.2.	Caracterización del ensayo	91
7.2.2.1.	Material utilizado	91
7.2.2.2.	Caracterización del modelo virtual	92
7.2.2.3.	Trabajo realizado	92
7.2.2.4.	Metodología	95
7.2.2.5.	Cierre del experimento	97
7.2.3.	Resultados	99
7.2.4.	Discusión	99
7.3.	Trabajo relacionado	101
7.4.	Conclusión y trabajo futuro	102
III	Cierre de la tesis	103
8.	Conclusiones	105
8.1.	Contribución	105
8.1.1.	Validación de hipótesis	106
8.1.2.	Limitaciones de las propuestas	107
8.2.	Divulgación de la contribución	108
8.2.1.	Publicaciones	108
8.2.2.	Actividades	109
8.2.2.1.	Ponencias	109

8.2.2.2.	Proyecto educativo de colaboración entre centros: Gemelo Digital: Didactificación de generación y uso de modelos virtuales de sistemas automatizados (2018/19)	109
8.2.2.3.	Colaboración en el Área de Innovación Tecnológica y Sistemas Inteligentes de Tknika, Centro de Investigación Aplicada de FP Euskadi (2019/Act.)	110
8.2.2.4.	Redacción de propuestas	110
8.3.	Competencias desarrolladas y conocimiento adquirido	111
8.3.1.	Capacitación preliminar	111
8.3.2.	Capacitación durante el desarrollo de la tesis	111
8.4.	Perspectivas y trabajo futuro	112
8.4.1.	Transferencia a industria	112
8.4.2.	Aplicación de los métodos propuestos	112
8.4.3.	Líneas de investigación futuras	113
IV	Bibliografía	115
	Bibliografía	117
V	Anexos	129
A.	Optimización energética en el marco de puesta en servicio virtual de sistemas industriales automatizados.	131
A.1.	Introducción	132
A.1.1.	El sobredimensionado de motores y la selección de consignas de variadores de velocidad	132
A.1.2.	Las posibilidades que ofrece la emulación	132
A.1.3.	Definición del problema	133
A.2.	Evaluación empírica	133
A.2.1.	Preguntas de investigación	133
A.2.2.	Caracterización del ensayo	134
A.2.2.1.	Material utilizado	134
A.2.2.2.	Caracterización del modelo virtual	134
A.2.2.3.	Trabajo realizado	139
A.2.2.4.	Metodología	140
A.2.2.5.	Cierre del experimento	144
A.2.3.	Resultados	144
A.2.4.	Discusión	144
A.3.	Trabajo relacionado	145
A.4.	Conclusión y trabajo futuro	145

Índice de figuras

1.1. Modelo virtual de un sistema de enrutado de producto.	5
1.2. Pirámide jerárquica de la automatización según la norma ISA 95. Fuente: [2].	7
1.3. Sistema dinámico de eventos discretos: célula de manufactura.	8
1.4. Modelado basado en una red de Petri: detalle de su estructura.	8
1.5. Modelado y programación de un GRAFCET.	8
1.6. Cronología de las revoluciones industriales. Fuente: [14].	10
1.7. Flujo de datos entre objeto físico y objeto digital. Fuente: [17].	12
1.8. Proyecto convencional vs. proyecto con puesta en servicio virtual.	13
1.9. Configuraciones SIL y HIL.	15
1.10. Gemelo digital en Simumatik Platform. Fuente: [36].	16
2.1. Modelado de una planta virtual. Fuente: [49].	24
2.2. Emulación de máquina de machos como soporte al proyecto de su reacondicionado. Fuente: [53].	27
4.1. Estructura de la tesis.	42
5.1. Célula de manufactura dotada de mesa lineal y robot.	50
5.2. Diagrama explicativo de la célula de manufactura.	50
5.3. Gemelo digital de la célula de manufactura.	50
5.4. Configuración del sistema de control según condiciones de trabajo asignadas.	54
6.1. Gemelo digital de un sistema dinámico de eventos discretos: paso de peatones regulado por semáforos.	60
6.2. Estructura de la red de Petri para el paso de peatones regulado por semáforos.	61
6.3. Implementación basada en redes de Petri: ordenador y PLC.	67
6.4. Herramienta de modelado y análisis de redes de Petri.	69
6.5. Software para la conversión de formato de los datos.	69
6.6. Gemelo digital de la célula de manufactura.	70
6.7. Configuración del sistema de control.	71
6.8. GRAFCET para la célula de manufactura.	72
6.9. Flujo de programa del algoritmo.	73
6.10. Fases de la metodología propuesta para la implementación de una red de Petri en un PLC.	75
6.11. Estructura de la red de Petri para la célula de manufactura.	76
6.12. Detalle de grafo de marcados accesibles.	77
6.13. Representación matricial generada.	78
6.14. Detalle del código de la interpretación de una red de Petri: evaluación de condiciones.	79
6.15. Medición del período de ejecución en Sysmac Studio.	80

6.16. Medición del período de ejecución en TIA Portal.	80
6.17. Detalle de la mejora introducida en el código de la interpretación de una red de Petri: evaluación de condiciones.	83
7.1. Simulador de camión. Fuente: [141].	86
7.2. Interacción entre los agentes de ADAPT.	89
7.3. Definición de una restricción asociada a un paso.	94
7.4. Restricciones asociadas a un paso.	94
7.5. Configuración del sistema de control en el testado de la plataforma: posible aplicación como herramienta de entrenamiento de personal novel.	96
7.6. Configuración del sistema de control en la fase operativa del proceso físico: posible aplicación como herramienta de diagnóstico de actividad.	96
7.7. Secuencia incorrecta, y exceso de duración de un paso.	98
7.8. Exceso de duración de un paso.	98
7.9. Secuencia correcta, pero exceso de duración total.	98
7.10. Interpretación en tiempo de ejecución.	99
A.1. Consumo energético debido a motores de inducción respecto al total de la demanda en países industrializados. Datos extraídos de [152] y [153].	131
A.2. Gemelo digital del sistema a automatizar.	135
A.3. Contador de energía en el sistema emulado.	136
A.4. Detalle del editor de componentes: comportamiento del servomotor.	137
A.5. Servomotor en el sistema emulado.	138
A.6. Configuración del sistema de control.	139
A.7. Puesta en servicio virtual del sistema automatizado.	141
A.8. Configuración base para puesta en servicio virtual.	141
A.9. Integración de un módulo de optimización.	141
A.10. Fases de la metodología.	142
A.11. Configuración de un test en la interfaz de usuario.	144
A.12. Resultados de un test, consistente en dos ciclos, en la interfaz de usuario: consumos estimados.	144

Índice de tablas

5.1. Señales de control en el modelo virtual.	51
5.2. Caracterización del experimento: asignación de condiciones de trabajo.	53
5.3. Resultados del experimento.	55
6.1. Interpretación a incorporar a la estructura de la red de Petri para el paso de peatones regulado por semáforos.	61
6.2. Interpretación a incorporar a la estructura de la red de Petri para la célula de manufactura.	77
6.3. Períodos de ejecución para las CPUs utilizadas y las implementaciones realizadas.	80
7.1. Pasos y situaciones a implementar.	93
7.2. Caracterización de los subsistemas en el caso de uso: situación de operación en modo automático.	100

Acrónimos

ADAPT	<i>Automatic Diagnosis of Activity and Processes in auTomation environments.</i>
CBM	modelado basado en restricciones o <i>Constraint-Based Modeling.</i>
CEO	director ejecutivo o <i>Chief Executive Officer.</i>
CFGS	Ciclo Formativo de Grado Superior.
CIFP	Centro Integrado de Formación Profesional.
CNC	control numérico computerizado o <i>Computer Numerical Control.</i>
CODESYS	<i>COntroller DEvelopment SYStem.</i>
CP LAB	laboratorio ciber-físico o <i>Cyber-Physical LAB.</i>
CPN	red de Petri coloreada o <i>Colored Petri net.</i>
CPS	sistema ciber-físico o <i>Cyber-Physical System.</i>
DEDS	sistema dinámico de eventos discretos o <i>Discrete Event Dynamic System.</i>
DES	simulación por eventos discretos o <i>Discrete Event Simulation.</i>
DM	modelo digital o <i>Digital Model.</i>
DS	sombra digital o <i>Digital Shadow.</i>
DT	gemelo digital o <i>Digital Twin.</i>
EEES	Espacio Europeo de Educación Superior.
ERP	planificación de recursos empresariales o <i>Enterprise Resource Planning.</i>
FMI	<i>Functional Mock-up Interface.</i>
FP	Formación Profesional.
GRAFCET	gráfico funcional de control de etapas y transiciones o <i>GRAPhe Fonctionnel de Commande Etape Transition.</i>
HIL	<i>Hardware-In-the-Loop.</i>
HiS	Högskolan i Skövde.
HMI	interfaz humano máquina o <i>Human-Machine Interface.</i>
IDLE	entorno integrado de desarrollo y aprendizaje o <i>Integrated Development and Learning Environment.</i>
IoT	Internet de las cosas o <i>Internet of Things.</i>
IS	sistema inteligente o <i>Intelligent System.</i>
IVC	<i>Integrated Virtual Commissioning.</i>
IVPC	<i>Integrated Virtual Preparation and Commissioning.</i>
LD	diagrama de contactos o <i>Ladder Diagram.</i>

MAS	sistema multiagente o <i>Multi-Agent System</i> .
MDPI	<i>Multidisciplinary Digital Publishing Institute</i> .
MES	sistema de control de producción o <i>Manufacturing Execution System</i> .
MQTT	<i>Message Queuing Telemetry Transport</i> .
MU-GEP	Mondragon Unibertsitatea - Goi Eskola Politeknikoa.
OEP	<i>Open Emulation Platform</i> .
OM	módulo de optimización u <i>Optimization Module</i> .
OP	objetivo parcial.
OPC UA	<i>Object Linking and Embedding (OLE) for Process Control Unified Architecture</i> .
P/T	Lugar/Transición o <i>Place/Transition</i> .
PC	ordenador o <i>Personal Computer</i> .
PDI	Personal Docente e Investigador.
PHM	<i>Prognostics and Health Management</i> .
PIPE2	<i>Platform Independent Petri net Editor 2</i> .
PLC	controlador lógico programable o <i>Programmable Logic Controller</i> .
PLM	gestión del ciclo de vida del producto o <i>Product Lifecycle Management</i> .
PN	red de Petri o <i>Petri Net</i> .
PrT-net	red Predicado-Transición o <i>Predicate/Transition net</i> .
PyME	Pequeña y Mediana Empresa.
RG	Grafo de marcados accesibles o <i>Reachability Graph</i> .
RQ	pregunta de investigación o <i>Research Question</i> .
S3D	Simumatik3D.
SCADA	sistema de supervisión, control y adquisición de datos o <i>Supervisory Control And Data Acquisition</i> .
SFC	diagrama funcional secuencial o <i>Sequential Function Chart</i> .
SIL	<i>Software-In-the-Loop</i> .
ST	texto estructurado o <i>Structured Text</i> .
UDT	tipo definido por el usuario o <i>User-Defined Type</i> .
UHS	ULMA Handling Systems S.Coop..
UPTC	ULMA Packaging Technological Center S.Coop..
VC	puesta en servicio virtual o <i>Virtual Commissioning</i> .
VSD	variador de velocidad o <i>Variable-Speed Drive</i> .

Parte I

Enmarque de la tesis

Capítulo 1

Introducción

Este documento está estructurado en tres partes o bloques principales, que a su vez constan de uno o varios capítulos.

Se abre esta Parte I, la de enmarque de la tesis, con el Capítulo 1, al que se da inicio con la motivación principal y el alcance la investigación realizada por el doctorando, y final introduciendo los antecedentes técnicos relativos al área de interés. El Capítulo 2 contiene una revisión del estado del arte, la cual desemboca en la definición del trabajo a realizar. Los objetivos e hipótesis enunciados, los materiales requeridos y los métodos aplicados son detallados en el Capítulo 3.

La Parte II corresponde al desarrollo de la tesis. El Capítulo 4 sirve de introducción a los experimentos detallados en los Capítulos 5, 6 y 7, y en el Anexo A. Contienen todo lo relativo al trabajo realizado, así como las contribuciones, su divulgación, y los resultados alcanzados.

El Capítulo 8 forma la Parte III, última de este documento. Contiene las conclusiones y la identificación de líneas futuras de investigación.

1.1. Motivación y alcance de la investigación

La tesis *Aplicación de tecnologías de digitalización a sistemas industriales automatizados basados en controladores lógicos programables* se enmarca en el Grupo de Investigación de Robótica y Automatización de [Mondragon Unibertsitatea - Goi Eskola Politeknikoa \(MU-GEP\)](#) [1]. En la misma se trata temática de interés compartido entre empresas fabricantes de maquinaria de la Corporación MONDRAGON, como por ejemplo [ULMA Handling Systems S.Coop. \(UHS\)](#) o [ULMA Packaging Technological Center S.Coop. \(UPTC\)](#), y la propia Universidad, tanto en su actividad de I+T y prestación de servicios a dichas empresas industriales como en el ámbito académico, de cara a preparar futuros profesionales.

El contexto de este trabajo es el de los sistemas industriales basados en [controlador lógico programable o Programmable Logic Controller \(PLC\)](#), concretamente en la problemática a la que se enfrenta el personal técnico en proyectos de soluciones automatizadas. Los avances tecnológicos y la creciente complejidad de las instalaciones colisiona con el interés por desarrollar soluciones de bajo coste y con plazos de entrega rápidos, tanto para nuevas implementaciones como para el reacondicionado de ya existentes. Las herramientas de emulación, con las cuales se experimenta en los casos presentados en este documento, pueden contribuir a acortar plazos a partir tanto de la agilización de la fase de puesta en servicio como a la sistematización

del desarrollo previo. Asimismo, se explotan los beneficios del uso de tecnologías de digitalización en otras fases del ciclo de vida del sistema.

Convencionalmente, la programación del dispositivo de control se apoya en simulaciones basadas en el forzado de entradas en su propio entorno de desarrollo, y la observación del comportamiento de las salidas, lo cual requiere un amplio conocimiento de la parte operativa a controlar y un adecuado protocolo de pruebas. El personal técnico de I+T a cargo de la experimentación con nuevas aplicaciones, los desarrolladores de proyectos para clientes y los centros educativos en los que se forman los profesionales del futuro se enfrentan a problemáticas similares:

- Por un lado, en proyectos de investigación en los que no necesariamente se dispone del equipamiento a automatizar, la validación de sistemas de control para nuevos desarrollos se reduce a la forma de proceder mencionada.
- Por otro lado, para implementaciones completas con equipamiento real, las pruebas son llevadas a cabo en el mismo, bien en instalaciones de la empresa desarrolladora, bien en las de la empresa cliente. Un software con errores puede poner en riesgo equipamiento, dispositivos y personas, y acarrear gastos imprevistos, quedando en evidencia el prestigio de la compañía fabricante, que se expone a la pérdida de pedidos futuros. Asimismo, y una vez operativa la solución automatizada, tareas como el ajuste de velocidades de motores eléctricos o el entrenamiento de personal inexperto pueden, al realizarse en el propio sistema físico, acarrear ciertos inconvenientes:
 - Los motores se encuentran generalmente sobredimensionados y acostumbran, además, a operar al 100 % de su capacidad nominal, con el consiguiente despilfarro energético, existencia de cuellos de botella y necesidad de almacenaje intermedio. Cualquier ajuste de parámetros operativos posterior a la puesta en servicio del sistema de control requiere disponibilidad del equipamiento.
 - Una acción formativa puede provocar paradas en la producción, generar defectivo o comprometer la seguridad de materiales y personas.
- Mientras, en el ámbito académico se ensayan las tareas y actividades descritas, dándose también los problemas derivados de las mismas. En ejercicios básicos simulados resulta complicado reproducir las señales que se darían en el proceso real. En implementaciones completas el alumnado trabaja contra maquetas que representan maquinaria industrial. El personal docente organiza el acceso al material de manera rotativa y limitada en tiempo, por el precio del equipamiento, la escasa cantidad de unidades disponibles y el reducido espacio en laboratorios. Si a ello se suma la mencionada dificultad para simular en el entorno de programación todos los escenarios que se pueden dar en el proceso, la puesta en marcha del proyecto se convierte en un banco de pruebas. Los errores en el código pueden derivar en situaciones de peligro y problemas de desgaste y/o rotura de elementos, lo cual se agrava cuando hay mecanismos sensibles y que requieren ajustes frecuentes, reduciéndose aún más la disponibilidad de la maqueta. Finalmente, y dada la dificultad para adquirir competencias de desarrollo de software de control en el aula, la dedicación a ellas es amplia y se dejan de lado otras de interés como las relativas al ajuste de parámetros operativos y la eficiencia energética.



FIGURA 1.1: Modelo virtual de un sistema de enrutado de producto.

Las tecnologías de digitalización asociadas a la Industria 4.0, como el [gemelo digital](#) o *Digital Twin* (DT) (Figura 1.1), traen consigo un cambio de paradigma en el ámbito descrito. La aparición y expansión de herramientas de emulación de sistemas industriales, unida a la creciente capacidad de computación de equipos informáticos, hace posible el testado de código de sistemas de control en un DT antes del montaje del equipamiento. Pero la aportación de un modelo virtual que emule una parte operativa va más allá que la validación virtual de programas previa a la puesta en servicio de la instalación. Se extiende a todo el ciclo de vida útil de la solución automatizada, haciendo posibles aplicaciones de entrenamiento de personal sin experiencia, la gestión del mantenimiento, y la optimización de procesos para su funcionamiento en términos eficientes tanto en términos operativos como energéticos. Esto último y la sostenibilidad forman uno de los pilares de la Industria 4.0. Por contra, el potencial que muestran las tecnologías de digitalización aplicadas a sistemas industriales automatizados contrasta con la escasez de personal dotado de las competencias técnicas necesarias para dar respuesta a la complejidad y el esfuerzo asociado al desarrollo de un modelo virtual. Esto se traduce en un coste, en tiempo y dinero invertidos, a lo que añadir el gasto en licencias de herramientas de emulación. La implantación de estas tecnologías no es, en consecuencia, generalizada en empresas industriales, así como en la propia [MU-GEP](#).

Para abordar la problemática descrita, en esta tesis se analiza el aporte de las herramientas de emulación al desarrollo, testado y validación de sistemas de control, a partir de la experimentación en varias actividades y fases del ciclo de vida útil de la solución automatizada. Se afrontan retos como los siguientes:

- Estructurar un marco de trabajo con emuladores de sistemas industriales automatizados para explotar los beneficios de estas tecnologías a lo largo de las distintas fases del ciclo de vida de dichos sistemas.
- Generar actividad docente e investigadora que aporte valor añadido a alumnado y entorno industrial.
- Contribuir al desarrollo de personal competente en este ámbito y hacer que el desarrollo de modelos virtuales sea menos costoso.

- Apoyar nuevas implementaciones en la actividad investigadora, que serían complejas de llevar a cabo al requerir pruebas contra procesos específicos pero no disponibles físicamente.
- Hacer uso de herramientas preferentemente libres e independientes de ningún fabricante de dispositivos de control.

1.2. Antecedentes

A continuación se definen conceptos clave de cara a que la persona lectora se familiarice con los sistemas industriales automatizados (sección 1.2.1) y la problemática a la que se enfrenta el personal técnico en la implementación de un [sistema dinámico de eventos discretos o *Discrete Event Dynamic System \(DEDS\)*](#) en un PLC (sección 1.2.2), concretamente en su testado y validación (sección 1.2.3). Se revisan términos relativos a las tecnologías de digitalización (sección 1.2.4) de sistemas industriales automatizados y el cambio de paradigma que originan tanto en el testado y validación de sistemas de control como en otras fases a lo largo de todo el ciclo de vida de la solución automatizada (sección 1.2.5). Finalmente, se introducen aspectos correspondientes a la conectividad entre dispositivos de control y procesos emulados (sección 1.2.6), y se presentan algunas herramientas software (sección 1.2.7), principalmente las utilizadas en los experimentos realizados.

1.2.1. Sistemas industriales automatizados

La industria es la parte de una economía que produce bienes materiales altamente mecanizados y automatizados. La norma ISA 95 proporciona una jerarquía funcional en un paradigma de organización industrial de fabricación automatizada. Este modelo adopta la forma de una pirámide con las siguientes capas o niveles (Figura 1.2):

- Nivel 0 (inferior) o de proceso: dotado de sensores y actuadores para la recogida de datos de funcionamiento y la ejecución de órdenes, respectivamente.
- Nivel 1 o de control: incluye dispositivos de tipo [control numérico computarizado o *Computer Numerical Control \(CNC\)*](#), PLC y/o robot para el control de maquinaria.
- Nivel 2 o de producción: contiene interfaces tipo [interfaz humano máquina o *Human-Machine Interface \(HMI\)*](#) y/o [sistema de supervisión, control y adquisición de datos o *Supervisory Control And Data Acquisition \(SCADA\)*](#) para la adquisición de datos y la supervisión del proceso.
- Nivel 3 o de operación: plataforma para supervisar, monitorizar y controlar el proceso de producción de las fábricas, conectando al mismo tiempo máquinas, personas y sistemas. Es decir, un [sistema de control de producción o *Manufacturing Execution System \(MES\)*](#).
- Nivel 4 (superior): corresponde a un [planificación de recursos empresariales o *Enterprise Resource Planning \(ERP\)*](#), conjunto de sistemas de información que permite la integración de operaciones de una empresa, especialmente las que tienen que ver con la producción, la logística, el inventario, los envíos y la contabilidad.

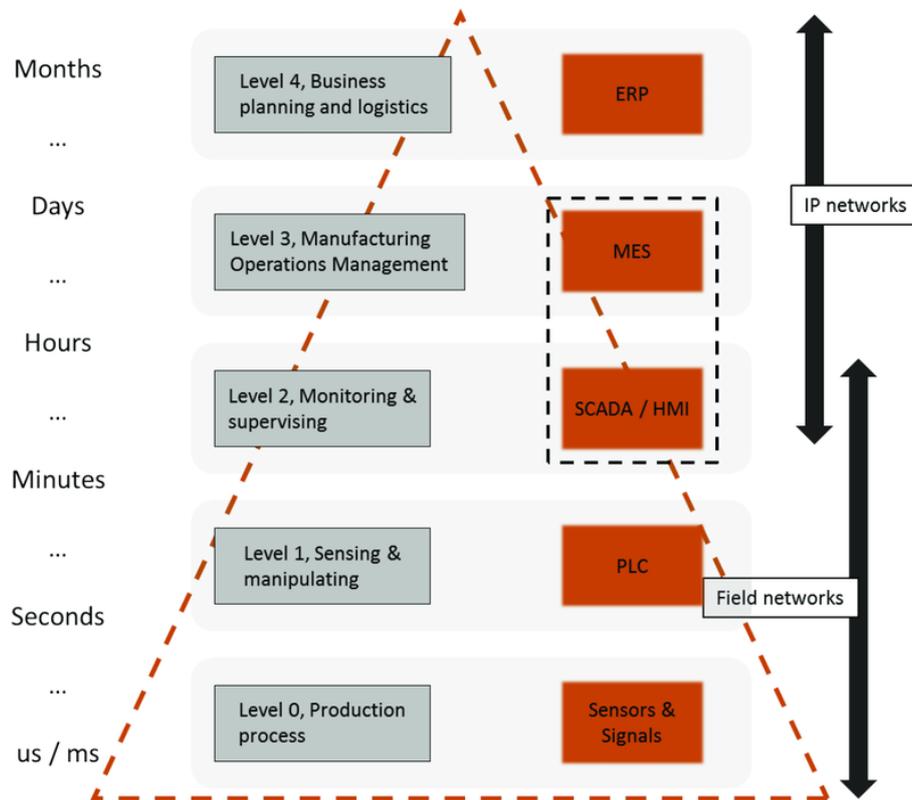


FIGURA 1.2: Pirámide jerárquica de la automatización según la norma ISA 95.

Fuente: [2].

1.2.2. Sistemas de control basados en PLC

El **PLC** es el dispositivo de control de uso más generalizado en entorno industrial. Está orientado a sistemas de tipo **DEDS**, y se basa en la información del proceso y del operador [3]. Su uso se debe a varias características que lo diferencian de un ordenador convencional [4]: (i) ejecución cíclica del programa; (ii) fiabilidad; (iii) adaptabilidad a la presencia de ruido eléctrico, vibraciones, temperaturas y humedad extremas; y (iv) fácil mantenimiento.

Un **DEDS** es un sistema dinámico y asíncrono, donde las transiciones de estado se inician por eventos que ocurren en instantes discretos de tiempo [5]. Ejemplos cotidianos de este tipo de sistemas son la sincronización de semáforos en un paso de peatones tras la expiración de la temporización de cada estado, o personas abandonando o incorporándose a una cola de espera a ser atendidas en un mostrador. En el ámbito industrial gran parte de los sistemas a automatizar son **DEDSs**: sistemas de fabricación flexibles y/o discretos dotados de robots y/o máquinas programables con evolución concurrente, sincronización y/o recursos compartidos. Sirva como ejemplo la Figura 1.3, la cual muestra una célula que consta de dos máquinas por las que las piezas pasan en el mismo orden, un almacén intermedio y un robot como recurso compartido, que podrá realizar una única operación de carga o descarga de una de las máquinas cada vez [6]. Para el modelado e implementación de **DEDSs** en **PLCs** se utiliza una de estas herramientas: **gráfico funcional de control de etapas y transiciones** o **GRAPhe Fonctionnel de Commande Etape Transition (GRAFCET)** y **red de Petri** o **Petri Net (PN)**.

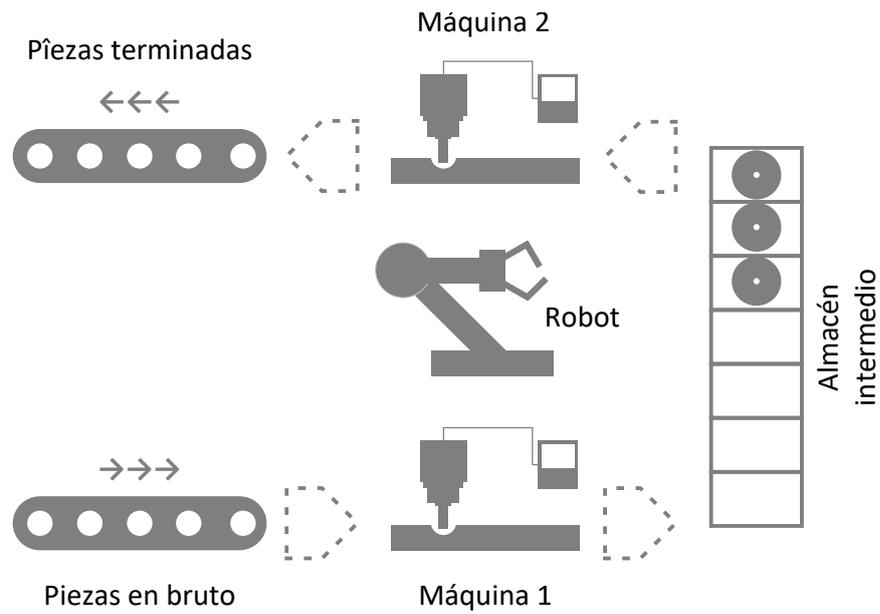


FIGURA 1.3: Sistema dinámico de eventos discretos: célula de manufactura.

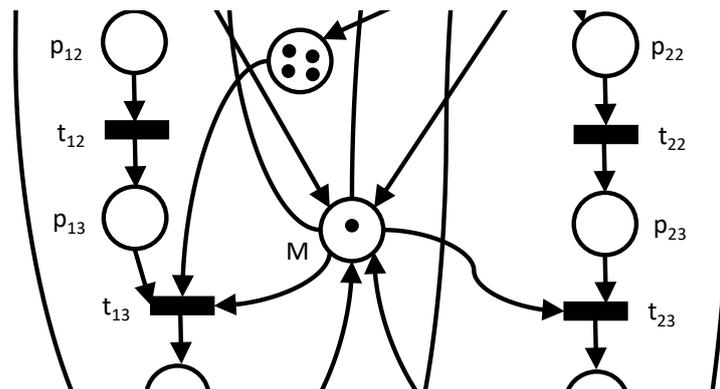
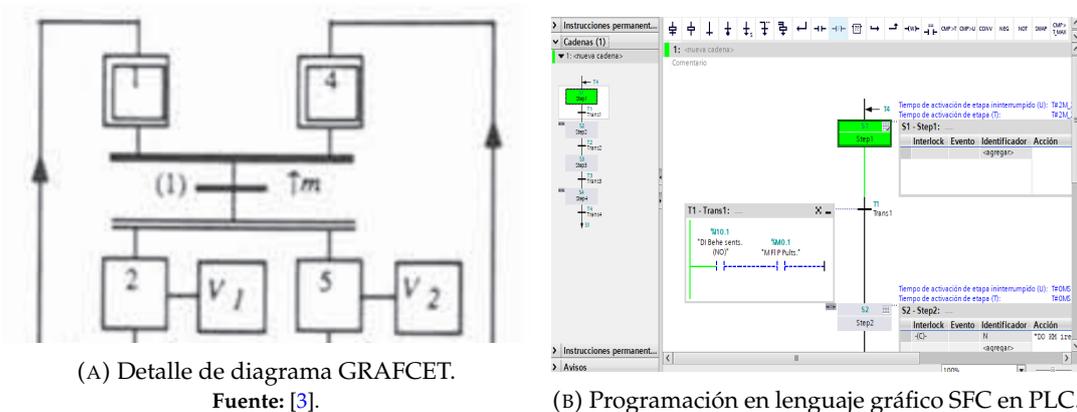


FIGURA 1.4: Modelado basado en una red de Petri: detalle de su estructura.



(A) Detalle de diagrama GRAFCET.

Fuente: [3].

(B) Programación en lenguaje gráfico SFC en PLC.

FIGURA 1.5: Modelado y programación de un GRAFCET.

La Figura 1.4 muestra parte de la estructura de una PN. Una PN es una herramienta de modelado gráfica y matemática aplicable a sistemas que se caracterizan por ser concurrentes, asíncronos, distribuidos, paralelos, no deterministas y/o estocásticos [7]. Herramienta gráfica por estar basada en un diagrama de bloques al que se añaden marcas para simular la dinámica del sistema, y matemática por la existencia de modelos algebraicos relativos a su comportamiento. Una PN es considerada como un método efectivo [8] para el diseño e implementación de DEDSs [9]. Se aplica en numerosos sectores, tales como la manufactura [10].

Un GRAFCET, caso particular de una PN, es una herramienta gráfica ampliamente utilizada para el modelado de DEDSs y su implementación en PLCs [3] (Figura 1.5). Se trata de una PN segura o binaria (Figura 1.5a), es decir, la propiedad de limitación de la red está garantizada siendo el posible marcado de cualquier lugar de la misma 0 o 1. Su implementación en dispositivos de control ha sido realizada asociando una variable booleana a cada etapa [11] [12], evaluándose su estado en cada ciclo de scan del PLC. En la actualidad los entornos de desarrollo incluyen un método gráfico de modelado y descripción de sistemas secuenciales adecuado para GRAFCET y estandarizado en IEC 61131-3 (Figura 1.5b), conocido como *diagrama funcional secuencial* o *Sequential Function Chart (SFC)*. En caso de no disponer de esta utilidad, se recurre a básculas set-reset en un *diagrama de contactos* o *Ladder Diagram (LD)* o se procede de manera equivalente en *texto estructurado* o *Structured Text (ST)*: (i) utilizando estructuras IF THEN, o (ii) programando la ecuación lógica de una báscula.

1.2.3. Testado y validación convencional de sistemas de control basados en PLC

Se acostumbra a testar el código de PLC a lo largo de su desarrollo y a partir del forzado entradas y/o variables, y la visualización de salidas y/o variables en el propio entorno del fabricante del dispositivo, sin visión directa del comportamiento del proceso. La validación de una solución automatizada suele realizarse durante la fase de puesta en servicio de la misma. Se hace uso del sistema real, incluso con el equipamiento ya montado, bien en las instalaciones de la empresa desarrolladora, bien en las de la compañía cliente.

Trabajar exclusivamente en el entorno de desarrollo del controlador, sin un procedimiento de test riguroso y un buen conocimiento del proceso a automatizar puede hacer que el sistema de control llegue incompleto o con errores a la puesta en servicio. Dicha fase final del proyecto puede tener una duración considerable, con gastos inesperados y/o innecesarios, un coste personal y social para el personal técnico a cargo, y posibles daños/lesiones en equipamiento/personas. La consiguiente pérdida de posicionamiento del fabricante en el mercado, al ser factores clave el tiempo de maquinaria parada en proyectos de reacondicionado y el plazo de entrega en nuevos desarrollos completos, puede afectar a su reputación y condicionar pedidos futuros. En definitiva, la puesta en servicio convencional no garantiza plazos de entrega cortos ni su cumplimiento.

1.2.4. Las tecnologías de digitalización en el marco de la Industria 4.0

En esta sección se contextualiza este trabajo en el marco de la cuarta revolución industrial, para posteriormente definir una serie de conceptos que se utilizan a lo largo del mismo.

1.2.4.1. La cuarta revolución industrial

Desde el inicio de la industrialización, los saltos tecnológicos han dado lugar a cambios de paradigma o revoluciones industriales (Figura 1.6): en los ámbitos de la mecanización (1ª revolución industrial), del uso intensivo de la energía eléctrica (2ª revolución industrial), y de la digitalización generalizada (3ª revolución industrial). En un contexto de digitalización avanzada en fábricas, [13] vislumbró un nuevo cambio de paradigma fundamental en la producción industrial, a partir de la combinación de las tecnologías de Internet y las tecnologías orientadas al futuro en el ámbito de los objetos inteligentes (máquinas y productos). Su visión de la producción futura contenía sistemas de fabricación modulares y eficientes, y caracterizaba escenarios en los que los productos controlarían su propio proceso de fabricación. Suponía que con ello se haría realidad la fabricación de productos individuales en un tamaño de lote de uno, manteniendo las condiciones económicas de la producción en masa.

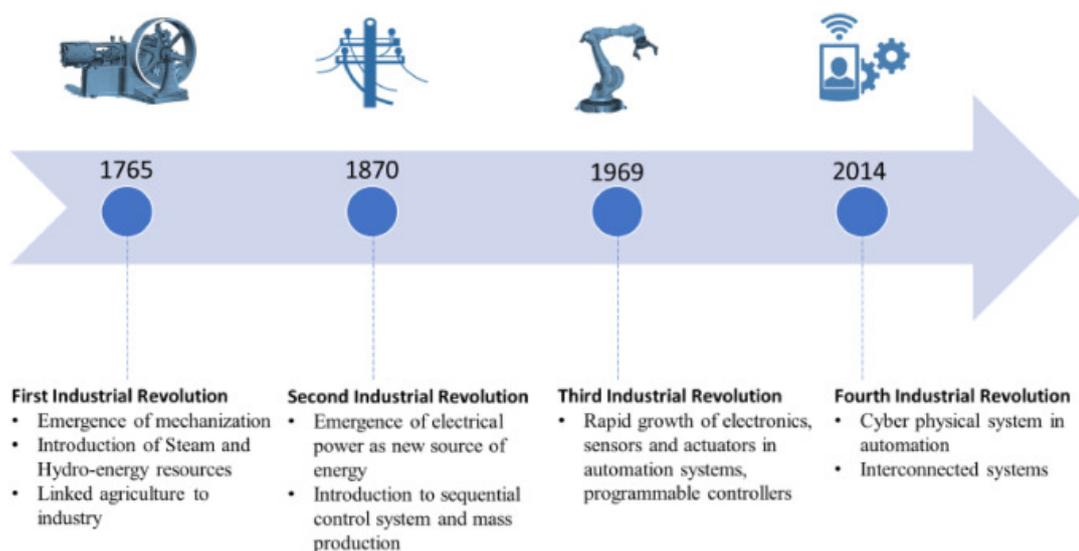


FIGURA 1.6: Cronología de las revoluciones industriales.

Fuente: [14].

Se estableció el término Industria 4.0 para una planificada 4ª revolución industrial, siendo el término una reminiscencia del versionado de software. Dicha revolución, y la transformación digital subyacente, avanzan de forma exponencial según [15], y está reconfigurando el modo de vida y de trabajo de las personas. La convergencia entre los espacios físicos y digitales, está revolucionando la forma de gestionar las operaciones de producción. La exigencia de aumentar la productividad, mejorar la flexibilidad y la resistencia y reducir los costes en la fabricación de la Industria 4.0 exige nuevos paradigmas que se ajusten al cambio de la gestión de la producción y las operaciones [16].

En dicho contexto de enorme competitividad, fuerte personalización de producto y emergencia de componentes software [17], la aplicación de tecnologías de digitalización en sistemas de manufactura trae consigo una mayor productividad [18], una integración simple de componentes inteligentes [19], y el control y la monitorización en tiempo real de dispositivos y elementos ciber-físicos [20], así como la planificación virtual de producto y proceso a partir de análisis de datos por herramientas de simulación [21].

1.2.4.2. Conceptos de interés

La Industria 4.0 ha traído consigo una serie de términos en el ámbito de interés. En esta sección se definen los utilizados en esta tesis.

1.2.4.2.1 Sistema ciber-físico

Se puede describir un **sistema ciber-físico** o *Cyber-Physical System (CPS)*, concepto clave de las arquitecturas de la Industria 4.0, como un conjunto de dispositivos físicos, objetos y equipos heterogéneos que interactúan con un ciberespacio virtual a través de una red de comunicación [22], y que realizan operaciones complejas para alcanzar un objetivo [23].

1.2.4.2.2 Emulación vs. simulación

Las herramientas de emulación han originado un cambio de paradigma en el desarrollo, testado y validación de soluciones automatizadas, tanto nuevas como de reacondicionado a la Industria 4.0, y para las cuales se demanda una fuerte personalización y plazos de entrega cortos. Conviene diferenciar las herramientas de emulación de las de simulación [24].

Herramientas de simulación

Las herramientas de simulación proporcionan un entorno diseñado para aproximarse al mundo real con el objetivo de contar con un ahorro computacional suficiente para la ejecución del modelo a mayor velocidad que en tiempo real. Intentan proporcionar un comportamiento similar al del sistema real. Se utilizan simplificaciones o aproximaciones para obtener una idea de cómo reacciona el sistema y analizar sus resultados. No necesita un sistema de control.

Herramientas de emulación

La emulación debe garantizar la fidelidad funcional. Intenta imitar el comportamiento de un sistema para realizar el mismo trabajo y producir los mismos resultados. Se replica el mundo real con suficiente concreción para que el equipo no pueda distinguirlo del mismo. El modelo emulado necesita que el sistema de control real funcione correctamente y, en consecuencia, puede ser utilizado para probar la funcionalidad o la lógica de un PLC, o un programa de robot.

1.2.4.2.3 Gemelo digital

Según la revisión de la literatura de [17] acerca del DT en la manufactura, se trata de una de las tecnologías de digitalización con mayor potencial en numerosos campos en la industria [25], y consiste en el equivalente virtual e informatizado de un sistema físico. Se trata esencialmente de un sistema funcional de optimización continua del proceso que está formado por la cooperación de las líneas de producción físicas con una especie de copia digital [26].

Más allá de la definición común de esa equivalencia digital de objetos físicos, con frecuencia se utilizan como sinónimos los términos **modelo digital** o *Digital Model (DM)*, **sombra digital** o *Digital Shadow (DS)*, y el ya referido y de uso generalizado DT. La diferencia radica en nivel de integración de datos entre la parte física y la digital (Figura 1.7):

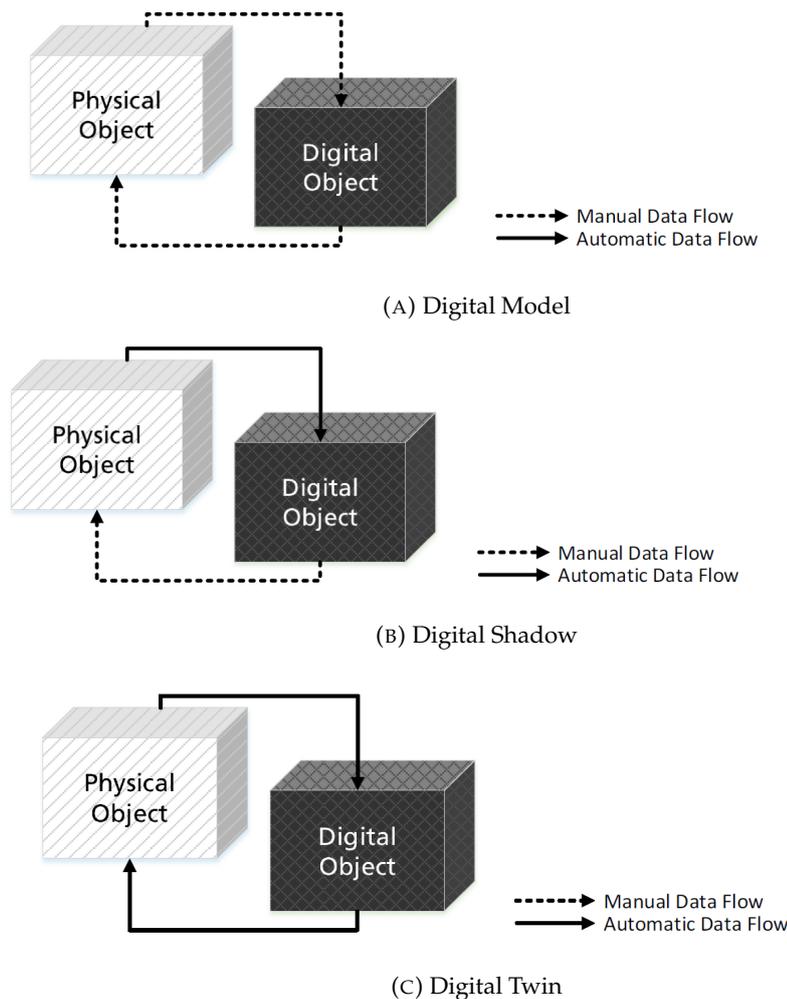


FIGURA 1.7: Flujo de datos entre objeto físico y objeto digital.

Fuente: [17].

- **DM**: es una representación digital de un objeto físico existente o previsto que no utiliza ninguna forma de intercambio de datos automatizado entre el objeto físico y el objeto digital (Figura 1.7a). Los datos digitales de los sistemas físicos existentes pueden seguir utilizándose para el desarrollo de dichos modelos, pero todo el intercambio de datos se realiza de forma manual. Un cambio de estado en cualquiera de ellos no tiene ningún efecto directo en el otro.
- **DS**: existe un flujo de datos unidireccional automatizado entre el estado de un objeto físico existente y un objeto digital (Figura 1.7b). Un cambio de estado del objeto físico conduce a un cambio de estado en el objeto digital, pero no al revés.
- **DT**: el flujo de datos entre un objeto físico existente y un objeto digital está totalmente integrado en ambas direcciones, es decir, la representación digital está totalmente integrada con el intercambio de datos en tiempo real (Figura 1.7c). El objeto digital puede actuar también como instancia de control del objeto físico. También puede haber otros objetos, físicos o digitales, que induzcan cambios de estado en el objeto digital. Un cambio de estado del objeto físico conduce directamente a un cambio de estado del objeto digital y viceversa.

En esta tesis, al igual que en la literatura, se ha adoptado **DT** como término general, pese a no describirse ningún flujo bidireccional de datos entre objeto físico y objeto digital.

1.2.5. El soporte del gemelo digital en sistemas industriales automatizados

La aplicación de herramientas de emulación de sistemas industriales automatizados se ha extendido desde las pruebas a las que se sometía inicialmente el código de dispositivos de control a diferentes tareas a lo largo de la vida útil de la solución automatizada.

1.2.5.1. Testado y validación de código de PLC en un gemelo digital

Hasta la aparición de sistemas de emulación en el mercado, un proyecto convencional de automatización industrial (identificado como *A* en la Figura 1.8) ha constado de las siguientes fases: diseño; desarrollo del sistema de control; instalación; y puesta en servicio durante la cual se realizan el testado y la validación del sistema de control.

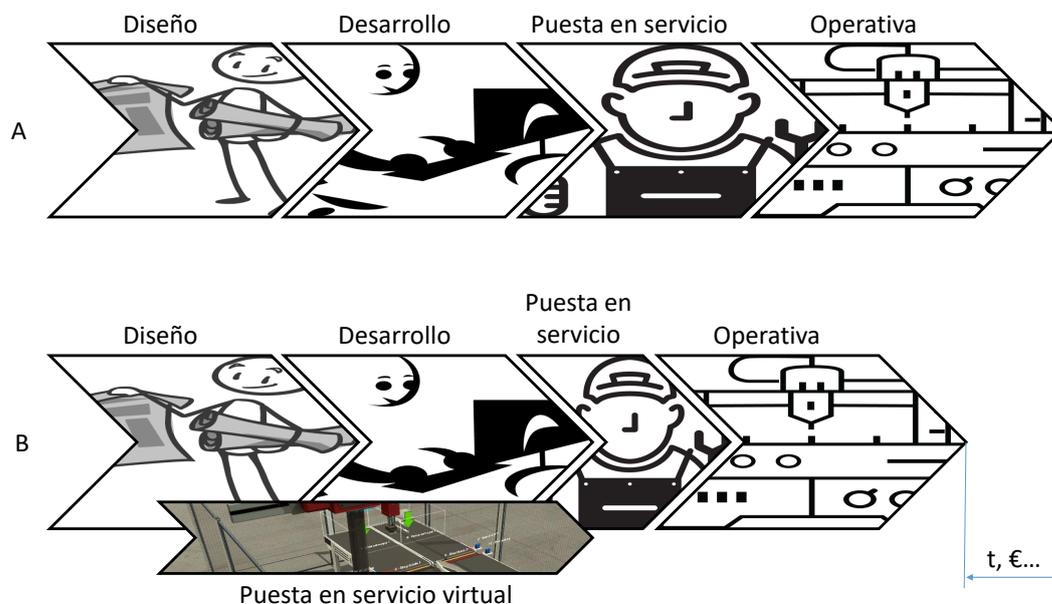


FIGURA 1.8: Proyecto convencional vs. proyecto con puesta en servicio virtual.

Un **DT** hace posible someter a pruebas los sistemas de fabricación y los programas de control asociados de manera virtual y previamente al montaje de la instalación real [27], lo que es conocido como **puesta en servicio virtual** o *Virtual Commissioning (VC)*. Se posibilita la realización de pruebas sin originar paradas y el testado de situaciones que en un sistema real puede resultar caras o complejas de reproducir [28], tanto en **PLCs** como en robots y sistemas de visión artificial.

Identificado como *B* en la figura, en un proyecto apoyado por la emulación el periodo de testado y validación virtual puede ir desde la fase de diseño a la de puesta en servicio. El manejo de un **DT** se refleja en la reducción de tiempo de puesta en marcha del sistema industrial automatizado. Se detectan comportamientos anómalos con antelación, y se reproducen situaciones que en el sistema físico serían costosas

o complejas [28], u ocasionar paradas en la producción. Estos beneficios, así como algunas debilidades, se tratan en la revisión de la literatura que tiene por objeto el Capítulo 2.

1.2.5.2. El gemelo digital más allá del testado y validación de código de PLC

El uso de la emulación a lo largo de todo el ciclo de vida de un sistema industrial automatizado es referido mediante dos términos similares: *Integrated Virtual Commissioning (IVC)* e *Integrated Virtual Preparation and Commissioning (IVPC)*.

IVC sugiere que la lógica de control debe ser testada continuamente contra un modelo digital, durante las diferentes fases de ingeniería [29], y no únicamente en paralelo a la construcción y puesta en servicio del sistema real. Incluso para la capacitación de personal novel. El modelo podría soportar otras actividades de ingeniería, más allá de las relativas a la programación, como el diseño y la construcción.

IVPC es referido por [30] como un marco para la preparación basada en modelos e implementación del sistema de control que admite pruebas con el hardware físico. La idea consiste en poder validar continuamente, utilizando modelos virtuales, el trabajo intermedio. Este debe ser el resultado del uso de métodos formales, durante la preparación de la producción y más tarde en la implementación del sistema de control.

1.2.6. Conectividad entre el dispositivo de control y el gemelo digital del sistema industrial automatizado

Object Linking and Embedding (OLE) for Process Control Unified Architecture (OPC UA) se ha convertido en el estándar para la comunicación entre los sistemas de control industrial y el software de terceros, como por ejemplo las herramientas de emulación. Basado en el principio cliente-servidor, es independiente del proveedor, es decir, permite conectar *ordenador o Personal Computer (PC)* y *PLC* de cualquier fabricante [31]. Esto hace posible probar los sistemas de control mediante un *DT* del proceso, leyéndose y escribiéndose de forma transparente variables del dispositivo.

Los *PLCs* de última generación incluyen conectividad *OPC UA*, considerado como el estándar de la Industria 4.0. Para aquellos controladores que no proporcionan esta característica se puede utilizar un software externo como Simumatik3D OPC UA Server v0.1.6, Matricon, UAExpert o Kepware, entre otros.

Para la conexión entre *PC* y *PLC* se puede seguir una de las configuraciones [32] de la Figura 1.9:

- Planta virtual y controlador virtual, o *Software-In-the-Loop (SIL)*: consiste en combinar un modelo preciso de la planta a controlar con un *PLC* emulado. Es una alternativa relativamente barata, pero al no utilizar equipamiento descuida aspectos relativos a la seguridad como colisiones de robots o daños personales.
- Planta virtual y controlador real, o *Hardware-In-the-Loop (HIL)*: se utiliza todo el hardware posible, como *PLC*, módulos auxiliares y dispositivos de campo. Una de sus ventajas es que se crea un entorno idéntico o muy similar al de la instalación real y el código se testa directamente en el dispositivo de control. Por el contrario, esta configuración supone un coste más elevado y es menos

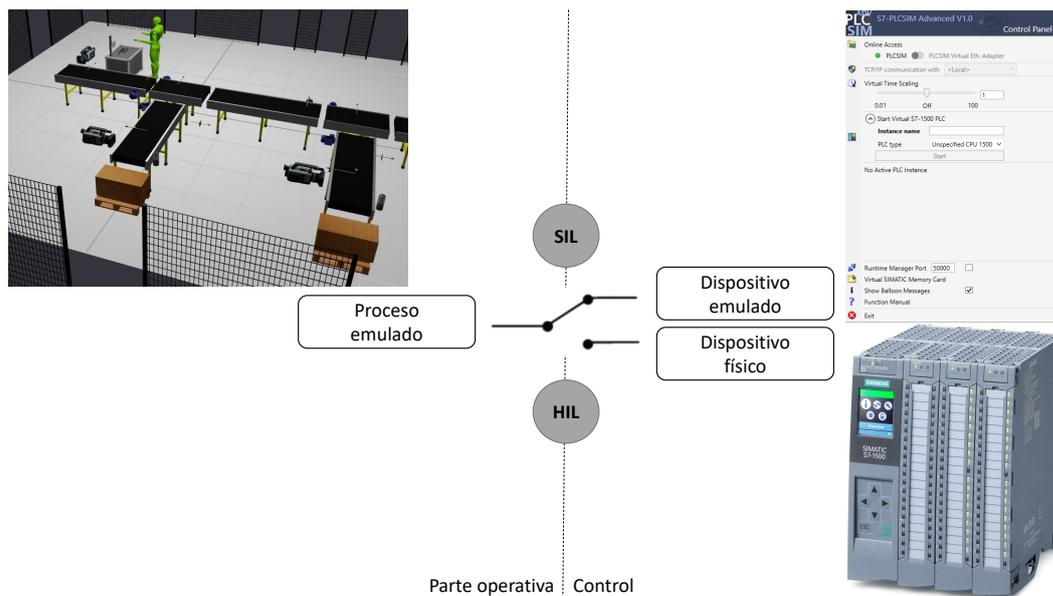


FIGURA 1.9: Configuraciones SIL y HIL.

flexible ante cambios durante la fase de diseño. Además, el riesgo de accidentes y lesiones es mayor.

1.2.7. Introducción a algunas herramientas de emulación disponibles en el mercado

Un software de emulación modela diferentes aspectos de los sistemas reales. Debe analizarse el uso y el resultado esperado del modelo para seleccionar la herramienta adecuada entre las disponibles. Esta tesis no busca someter a juicio herramientas para el desarrollo de DTs, sino explotar los beneficios de su uso en una serie de experimentos y desarrollos. Esta sección introduce brevemente al lector a software disponible en el mercado y sus particularidades.

SIMIT de Siemens, por ejemplo, es adecuada para la VC de aplicaciones de proceso, tales como fundiciones o plantas de tratamiento de aguas. A pesar de no modelar la geometría y la física del sistema, proporciona una interfaz de usuario amigable para el modelado del comportamiento, así como una buena conectividad para PLCs del propio fabricante. Otras plataformas como ABB Robot Studio, Excelgo Xperior, Simumatik3D (S3D) [33] o NX Mechatronic Concept Designer [34] proporcionan herramientas para el modelado de la geometría y la física del sistema. Ello las convierte en indicadas para aplicaciones de manipulación de material y células robóticas, entre otras. Tecnomatix Process Simulate es una herramienta intermedia entre las anteriores que ofrece la posibilidad de modelar la geometría y el comportamiento de los sistemas pero no incluye un motor de física [35].

Estas son las herramientas utilizadas a lo largo de este trabajo:

- CIROS Education 6.2 [37] es un entorno virtual de aprendizaje para sistemas controlados por PLC, con las mismas funciones que CIROS Studio, excepto la generación de nuevos modelos y la conexión de controles de robots. Ofrece un entorno de trabajo ideal para la programación con dispositivos Siemens y con controladores de otros fabricantes. Su extensa librería de modelos contiene DTs de estaciones de Festo Didactic [38], que pueden ser controlados desde un PLC

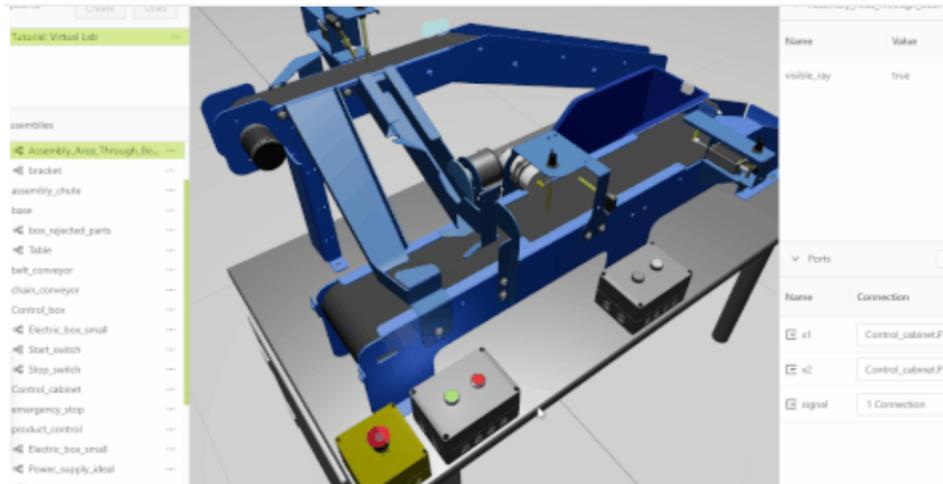


FIGURA 1.10: Gemelo digital en Simumatik Platform.

Fuente: [36].

virtual integrado, un PLC emulado en su propio entorno de desarrollo, o por hardware externo a través de EasyPort. MU-GEP cuenta con licencias y varias estaciones didácticas en laboratorios.

- Factory I/O v2.2.3 [39], emulación de fábrica en 3D orientada al aprendizaje de tecnologías de automatización, es sencilla de manejar. Permite construir con rapidez una sistema virtual a partir de componentes industriales comunes. Incluye escenas inspiradas en aplicaciones industriales típicas, con diferentes niveles de dificultad. Además de para formación relativa a la programación de PLCs, puede utilizarse con otros dispositivos y tecnologías. Es un entorno de creciente implantación y MU-GEP tiene interés en probar sus capacidades y desarrollar implementaciones virtuales basadas en las escenas disponibles. Dispone de un útil periodo de prueba gratuito.
- S3D v1.0.3, software de licencia gratuita para uso educativo, es muy intuitivo. Se trata de una herramienta de emulación desarrollada por Simumatik AB, compañía fundada en Suecia en 2014. Permite testar programas de PLCs y robots industriales de forma sencilla. Utilizado en cursos de automatización y proyectos de investigación de Högskolan i Skövde (HiS) y MU-GEP, entre otras universidades, en S3D es posible emular diferentes aspectos de sistemas mecatrónicos: geometría, física, cinemática y su comportamiento. Los modelos de emulación se construyen introduciendo componentes, como sensores, motores o cilindros, y conectándolos entre sí para crear un sistema.
- Simumatik *Open Emulation Platform (OEP)* o Simumatik Platform [36] es una evolución de S3D (Figura 1.10). Aprovecha el poder de la nube y es segura y accesible desde cualquier lugar y en cualquier momento. Sólo se necesita un dispositivo con conexión a Internet y un navegador para acceder al entorno. Por un lado, se puede acceder gratuitamente a DTs públicos. Por otro, la suscripción a una organización permite a los equipos trabajar con mayor eficacia, ya que se reduce el tiempo de modelado al crear cualquier componente mediante un editor y organizarlos en bibliotecas para poder reutilizarlos. Independiente del sistema de control, esta herramienta (i) ayuda a personal técnico a desarrollar y ofrecer soluciones automatizadas de forma eficiente a través de su VC, y (ii) mejora la experiencia de aprendizaje del alumnado en contenidos

relativos a control industrial (PLCs, robots, etc.) en un entorno de bajo coste, como propone [40]. Se dispone de licencias en MU-GEP y se cuenta con interés en probar e implementar este nuevo entorno.

Capítulo 2

Estado del arte

En un contexto de creciente complejidad de los procesos industriales, y en el que los avances tecnológicos obligan a dar por finalizado el ciclo de vida útil de maquinaria antigua o a su reacondicionado, se demandan soluciones automatizadas de bajo coste y plazos de entrega cortos. El software de emulación emerge para apoyar la validación de programas de PLCs o robots, agilizando así la puesta en servicio de maquinaria, así como otras tareas en otras fases del ciclo de vida de una instalación. En esta sección se repasan estudios de relevancia presentes en la literatura en lo que se refiere a la caracterización de DTs y su contribución al reto de agilizar el testado y la puesta en servicio de sistemas de control, así como su apoyo a un desarrollo sistemático de estos. La revisión se extiende a todo el ciclo de vida útil de la implementación, es decir, la evolución del concepto de VC a IVC o IVPC. Se citan artículos relacionados con esta problemática, discutiendo los resultados obtenidos hasta la fecha y desembocando en la búsqueda de oportunidades de investigación y la definición del trabajo a realizar en esta tesis.

La revisión realizada ha partido de la definición de palabras clave para los conceptos y áreas de interés y su aplicación, combinada con criterios de filtrado, en la búsqueda de literatura. A continuación se presenta lo extraído de este análisis.

2.1. Caracterización de la emulación de sistemas industriales automatizados

En esta sección se revisa la evolución del concepto de DT, las configuraciones más comunes y el estándar de uso general para las comunicaciones entre el sistema emulado en un PC y el sistema de control, y las principales técnicas de modelado.

2.1.1. El concepto de gemelo digital

Se atribuye el origen del DT a la presentación, por parte de Michael Grieves, de su trabajo con John Vickers de la NASA, en el transcurso de una conferencia sobre *gestión del ciclo de vida del producto o Product Lifecycle Management (PLM)* en 2003. Describiendo las representaciones virtuales de los productos como "relativamente nuevas e inmaduras" y los datos recogidos sobre los productos físicos como "limitados, extraídos manualmente y, en su mayoría, en papel" [41], Grieves y Vickers estimaron que un modelo virtual de un producto proporcionaría las bases para la PLM. A partir de la definición inicial de un DT como una representación virtual de un producto físico que contiene información sobre dicho producto, con sus orígenes en el campo citado, en un primer artículo Grieves amplía esta descripción a un conjunto de tres componentes [42]: un producto físico, una representación virtual de

ese producto y las conexiones de datos bidireccionales que alimentan los datos del físico a la representación virtual, y la información y los procesos de la representación virtual al físico.

Una definición más detallada y ampliamente reconocida en el ámbito de la industria de manufactura califica el DT como una simulación integrada multifísica, multiescala y probabilística de un producto complejo, y que utiliza los mejores modelos físicos disponibles, actualizaciones de sensores, etc., para reflejar la vida de su correspondiente gemelo [43] [25].

Mientras en su origen el concepto analizado describe la reproducción de un producto, el estado del arte [17] permite que los procesos (fabricación, generación de energía, etc.) sean también objeto de reproducción en el espacio virtual para obtener los mismos beneficios. La primera referencia al concepto ya como DT tiene lugar en el ámbito aeroespacial, en forma de hojas de ruta tecnológicas de la NASA [44]. Más allá de puros datos, los DTs incluyen algoritmos que describen su equivalente real y deciden sobre la acción en el sistema de producción a partir de estos datos procesados [21] [45].

En términos de fabricación, el DT consiste en una representación virtual de un sistema de producción que es capaz de funcionar en diferentes disciplinas de simulación que se caracteriza por la sincronización entre el sistema virtual y el real, gracias a los datos detectados y los dispositivos inteligentes conectados, los modelos matemáticos y la elaboración de datos en tiempo real. Su papel dentro de los sistemas de fabricación de la Industria 4.0 es explotar estas características para prever y optimizar el comportamiento del sistema de producción en cada fase del ciclo de vida en tiempo real [46].

La revisión de [17] resalta, además, el hecho de que únicamente el 18% de los casos analizados y presentados como DT lo son realmente en términos de flujo de datos entre parte física y parte digital, pese a su adopción como término general. Asimismo, trae a la luz aplicaciones como la planificación y el control de la producción [45], el mantenimiento [47] y la planificación de planta [18].

2.1.2. Conectividad entre el dispositivo de control y el gemelo digital

La literatura refleja el uso de modelos virtuales de sistemas industriales automatizados en configuraciones de tipo SIL, con el sistema de control emulado, y HIL, con el sistema de control físico. Sirvan como ejemplos de SIL trabajos como el de [48], que realiza la VC de una célula robótica, o el de [32], que comprueba en un aula académica la mejora en tiempo y calidad del trabajo que supone un proyecto soportado por DT. Aplicaciones de HIL son la plataforma de [49] para el aprendizaje del concepto de VC por parte de alumnado, personal docente e investigador y profesionales de industria; y la conexión, por parte de [50], de un sistema simulado en un ordenador a un PLC para la validación de programas en ejecución en este último. Además, [51] facilita el aprendizaje de programación de PLCs en un curso de automatización industrial mediante el uso de ambas configuraciones.

El uso del estándar OPC UA es generalizado. Constancia de ello dejan estudios como el de [50], que desarrolla una interfaz basada en Java; la propuesta y estudio de una metodología para el reequipamiento de maquinaria con CNC de [52]; o el reacondicionado de una máquina operativa desde principios de los 90 en la industria de automoción por parte de [53].

2.1.3. Técnicas de modelado

Las herramientas que se utilizan para la VC de sistemas automatizados están basadas en la emulación, considerada como la técnica más adecuada para el testado de un sistema de control [54] [31] en comparación con la simulación, ya que garantiza la fidelidad funcional. Se replica el mundo real con suficiente concreción para que el sistema de control no pueda distinguirlo del mismo [24].

Las técnicas que se aplican en el desarrollo de modelos virtuales son la de nivel E/S y la de nivel componente, siendo la primera de ellas la estrategia más común para la construcción de modelos de VC:

- Nivel E/S: este nivel de detalle es el adecuado para la validación de programas de PLC, ya que es el nivel mínimo requerido para la realización de pruebas relativas al software de control. Carece del potencial necesario para apoyar otras fases.
- Nivel componente: sirve, además, para apoyar el diseño o la optimización energética, entre otras tareas. Se modela individualmente cada componente a partir de aspectos geométricos, cinemáticos, funcionales y de interfaz, pudiendo ser utilizado sin necesidad de conocimientos adicionales a los de automatización.

El estándar de modelado más habitual es *Functional Mock-up Interface (FMI)*, que permite encapsular modelos para su posterior inclusión en un modelo de emulación completo junto con otros aspectos. Independiente de la herramienta, admite tanto el intercambio de modelos como la co-simulación de modelos dinámicos. Por ejemplo, [55] presenta un método para la construcción de modelos energéticos independientes siguiendo dicho estándar.

2.2. El reto de agilizar el desarrollo, el testado y la puesta en servicio de sistemas de control

Partiendo de la problemática a la que se enfrenta el personal técnico que participa en la puesta en servicio de soluciones automatizadas, en esta sección se identifican trabajos de relevancia en lo que al efecto del uso de la emulación para la validación de software de control se refiere, así como los intentos por generar este de forma sistemática y con menor probabilidad de contener errores.

2.2.1. Antecedentes: puesta en servicio convencional

Las empresas desarrolladoras de sistemas basados en PLC se enfrentan a un contexto de creciente complejidad y personalización del producto, avances tecnológicos que obligan a adaptar maquinaria antigua o dar por concluido su ciclo de vida útil, y una fuerte demanda de reducción de costes y plazos de entrega cortos. La validación de código de control se convierte en una fase crítica según [56], ya que tiene lugar en la maquinaria ya montada y en el tramo final del proyecto. La puesta en servicio puede suponer un 25 % de la duración total del proyecto [57], además de acarrear gastos no previstos e innecesarios, desperfectos, percances, etc. Esta fase es conocida como aquella en la que se gasta tiempo y dinero [58] y se pone en juego la reputación de la empresa fabricante, condicionando o incluso echando por tierra futuros pedidos [57].

En aquellos sistemas automatizados comandados por motores asíncronos trifásicos, generalmente sobredimensionados, emerge otra serie de problemas más allá de los relativos a la validación del código de control. Al obvio consumo excesivo de energía se le añade la necesidad de almacenamiento intermedio de material en aquellos puntos que resulten ser cuellos de botella. La aparición del **variador de velocidad** o *Variable-Speed Drive (VSD)* trae consigo la posibilidad de ajustar sus parámetros, tales como la consigna de velocidad, tras la puesta en servicio de la solución automatizada. Esto, al requerir disponibilidad de equipamiento, puede alargar plazos de entrega.

Poniendo el foco en las empresas cliente de proyectos de automatización industrial, [52] destaca la existencia de numerosa maquinaria operativa con necesidad de adaptación al nuevo paradigma de la Industria 4.0. El tiempo de puesta en servicio, con la producción detenida, se antoja especialmente crítico. Asimismo, los plazos de entrega de instalaciones nuevas suponen un diferencial a la hora de elegir empresa proveedora en el mercado [59].

Otro de los agentes clave en este escenario son las compañías fabricantes de soluciones automatizadas. La detección de errores durante el desarrollo de código de dispositivos de control no es sencilla. Los entornos de desarrollo de **PLCs** incluyen herramientas de simulación en las que se fuerzan entradas y se observan salidas del mismo. Resulta necesario un buen conocimiento del sistema a automatizar, así como un adecuado procedimiento de test para no dejar aspectos de las especificaciones de la aplicación sin contemplar. Los programas de control pueden llegar a la fase de validación en el sistema real con una maduración insuficiente.

En consecuencia, y desde la óptica de proveedores, clientes, y personal del ámbito académico, la puesta en servicio convencional no es considerada como una forma de proceder eficaz, no da respuesta adecuada a la necesidad de resultados de calidad, reducción de costes y plazos de entrega cortos.

2.2.2. La irrupción del gemelo digital: puesta en servicio virtual y otras aplicaciones

La literatura no es ajena al cambio de paradigma originado por la aparición de herramientas de emulación tanto inicialmente en el desarrollo, testado y validación de código de dispositivos de control, como posteriormente a lo largo del ciclo de vida de la solución automatizada.

2.2.2.1. Puesta en servicio virtual basada en un gemelo digital

Un proyecto de automatización apoyado por la **VC** ve reducido el tiempo de test y validación del sistema de control [57]. Se prueba e integra este previamente a que el sistema de fabricación se encuentre completamente instalado o modificado, es decir, antes de la puesta en servicio real, al utilizarse un modelo virtual de dicho sistema [53]. Se detectan con antelación comportamientos no deseados y se pueden llevar a cabo tests, tanto relativos a **PLCs** como de sistemas robóticos y de visión artificial, sin provocar paradas y que podrían ser caros o complejos de llevar a cabo en un sistema real [28]. Para ello, los resultados de la simulación, que puede ser multi-nivel [53] (proceso, célula, línea), deben predecir que los resultados reales serán útiles, para lo que [60] introduce el concepto de confianza virtual. Trabajos como [49] y [32] muestran los principales beneficios de la **VC**: la reducción del tiempo de puesta en servicio y la mayor calidad de los resultados obtenidos. La mano de obra requerida

en dicha fase final del proyecto es cuantificada por [59] en un 50 % menos respecto a proyectos no soportados por la emulación. Por contra, autores como [58] sacan a la luz hándicaps como el esfuerzo requerido para el desarrollo del modelo y la falta de personal dotado de las competencias técnicas adecuadas.

La evolución histórica de las referencias al uso de modelos en la puesta en servicio de proyectos de automatización industrial está acotada a un período reciente. Realizándose, a fecha de noviembre de 2018, una búsqueda del término *virtual commissioning*, entrecomillado y en minúscula, en la base de datos bibliográfica Scopus¹, se llega a los resultados que se sintetizan a continuación. Acotando dicha búsqueda a campos representativos como título, abstract y palabras clave, y a las áreas de ingeniería e informática, se concluye que la primera mención explícita al concepto de VC data de 2004. La evolución del número de publicaciones es ligeramente ascendente y se hace notoria a partir de 2015, con cerca de 30 por año. Por países, más de la mitad de los autores y las autoras son de Alemania. Ocho de los diez países con más aportaciones son europeos, siendo los dos restantes EEUU y Corea del Sur. Previamente, a finales de los 90, ya se reclaman modelos y se hace referencia a combinaciones de los términos *visual* (o *virtual*) y *validation* (o *verification*), entre otros conceptos. Por ejemplo, [61] introduce el *soft-commissioning*, y realizando una búsqueda de este se llega a un único resultado, donde se verifica un PLC mediante *simulación por eventos discretos* o *Discrete Event Simulation (DES)* [62]. En las referencias bibliográficas de estos dos artículos se manejan las combinaciones ya mencionadas.

2.2.2.2. Otras aplicaciones del gemelo digital

Autores como [63] destacan la validez de un DT en el nuevo paradigma de la industria conectada. Combinando descripción física y funcional con datos operativos, un modelo virtual es de utilidad en otras fases del ciclo de vida de un sistema industrial automatizado, y no únicamente para la verificación del programa de control [64] [65]. Las referencias en la literatura al uso de modelos en otras fases son relativamente recientes, y no hacen necesariamente mención explícita a IVC o IVPC, que son marcos que se proponen para esta problemática. A fecha de noviembre de 2018, se realiza una búsqueda para cada uno de estos términos, entrecomillado y en minúscula, en la base de datos bibliográfica Scopus. Se acota la búsqueda a campos representativos como título, abstract y palabras clave, y a las áreas de ingeniería e informática. Por una parte, para IVC hay un total de tres resultados, todos ellos de autores alemanes. Datan de 2007 ([66]), 2013 ([67]) y 2014 ([29]). Por otro lado, para IVPC hay otros tres resultados, todos ellos del mismo autor sueco. Datan de 2016 ([30]) y 2017 ([28] y [68]). En [30] se propone el marco IVPC, con modelos de VC que se utilizan como base para la preparación y la implementación del sistema de control asistidos por métodos formales. Dicho marco es trabajado en las otras dos publicaciones. En [28] se presenta un método de modelado automático a partir de programas de robot reales y un modelo de célula robótica, con continuo análisis y validación. Finalmente, en [68] se hace uso de herramientas para VC y un software de planificación de secuencias para la simulación y actualización de robots y controles de células robóticas en entornos modificables.

Sin hacer uso de los términos mencionados, las referencias a casos de uso de modelos virtuales en distintas fases del ciclo de vida útil de maquinaria o procesos automatizados es considerable. Se han analizado publicaciones de casos de uso de

¹<https://www.scopus.com/>



FIGURA 2.1: Modelado de una planta virtual.

Fuente: [49].

emulación en varios ámbitos y fases. Modelos y datos de un DT facilitan la realización de nuevos servicios con un esfuerzo moderado, especialmente combinando modelos y medidas reales. Las búsquedas realizadas reflejan los beneficios del uso de modelos virtuales de planta desde la planificación del proyecto hasta la operativa normal de la instalación.

Entre los campos de aplicación identificados se encuentran:

- El proceso de ingeniería en su globalidad. Se detectan casos como la simulación, combinada con el software original de control, que permite mejorar y acelerar el proceso de ingeniería en su conjunto [66]; el diseño de una línea productiva analizando los resultados de las modificaciones sin realizarlas [69]; la simulación del flujo de material combinada con herramientas de VC [67]; o la extensión del uso del modelo a otras etapas de ingeniería, de manera que la prueba continua de programas hace mejorar tanto el modelo como el software de control [29].
- Aplicaciones formativas y/o de entrenamiento de personal de operación y mantenimiento en fábrica. Se reproducen tanto el funcionamiento normal como escenarios de fallo sin forzar paradas en la producción. Mientras [49] desarrolla una plataforma didáctica para VC en un laboratorio de Industria 4.0, minimizando los errores de programación de controladores (ver Figura 2.1), [51] hace uso de un aula virtual para un curso de automatización industrial. Las herramientas interactivas para entrenamiento en fábrica que presenta [70] se aplican en un caso de éxito ideado para plantas acereras y están basadas en HIL. El enfoque de [71] para la creación de modelos de realidad virtual permite (i) la simulación de planta real (por ejemplo, para formación) y (ii) operación remota (por ejemplo, en entornos peligrosos). Por último, hay referencias al aprendizaje de sistemas hidráulicos en entorno educativo [72].
- Diagnóstico y mantenimiento. La Industria 4.0 permite abordar problemas como la detección de fallos y anomalías en los sistemas industriales en tiempo real a través de tecnologías emergentes [73] como Internet de las cosas o Internet of Things (IoT) y el aprendizaje automático. La metodología modular para mantenimiento correctivo de [74], hace uso del DT de un módulo de automatización, donde una interfaz visual atribuye fallos y se realiza la acción correctora. Mientras, [26] habla de la optimización continua y el mantenimiento proactivo, mediante análisis en CPSs (dispositivos conectados al DT). El CPS

de recogida de datos de [75] ha permitido crear un conjunto de datos que puede utilizarse para entrenar modelos de aprendizaje automático de diagnóstico temprano de una enfermedad epidémica fúngica. El método *Prognostics and Health Management (PHM)* basado en un DT de 5 dimensiones de [76] es aplicado en equipamiento complejo de aeroturbinas, de difícil acceso y donde se dan vibraciones y una degradación progresiva. El método de modelado de [77] tiene por objeto reducir la probabilidad de fallo, y se aplica a un CNC de tres ejes con sensores de posición, vibración y velocidad, prediciendo vida útil y diagnosticando fallos (fenómeno, razón). La combinación de modelos virtuales y datos reales en un entorno coherente y alineado [45], hace posible la realización de predicciones realistas, apoyo a operadores, planificadores y personal de mantenimiento, de manera que el sistema productivo puede reaccionar autónomamente ante pedidos, cambios de prioridades o perturbaciones. Finalmente, [63] trabaja un caso de uso para el mantenimiento de una máquina de puntos para operar desvíos ferroviarios. En resumen, las referencias al uso de un DT para mantenimiento [78], predicción de vida útil en aplicaciones como la fabricación aditiva [79] y toma de decisiones [80] son numerosas.

- Dimensionado de componentes, ajuste de parámetros y optimización del funcionamiento y/o del consumo energético: respecto a esto último, cabe destacar que para un país industrializado como Alemania, [81] cuantifica en un 50 % el consumo energético de la industria respecto a lo que se genera, suponiendo un 70 % del mismo el dedicado a accionar motores eléctricos. En HiS se ha desarrollado un método para la optimización energética de una célula de fabricación ficticia [82] a partir del ajuste de velocidades de los procesos más rápidos a la del cuello de botella de la instalación. En lo referente a la optimización, [83] propone una arquitectura y un método para extraer cambios en el proceso, retroalimentando modelos de ingeniería de cara a una mayor eficiencia en futuros diseños. Además, [84] presenta un ejemplo de aplicación de la simulación al diseño virtual y optimización de una línea de producción con un enfoque integrado para el ajuste de parámetros y [18] hace uso de un DT como herramienta para el ajuste de parámetros a partir de la adquisición automática de datos de proceso. Por último, y con sendas aplicaciones en casos industriales en el sector ferroviario, la arquitectura de DS de [85] para la optimización de líneas y células de producción puede ser utilizada en las fases de diseño y funcionamiento, y el enfoque de [86], basado en la integración de un DT y un método multiobjetivo, planifica a corto plazo la producción de sistemas multi-producto.

2.2.3. Intentos de sistematización del desarrollo de código

Conscientes de que minimizar errores en el código puede traer consigo puestas en servicio más ágiles y plazos de entrega competitivos, no son pocos los estudios dedicados a la fase de desarrollo del mismo. Aun contando en la actualidad con DTs para la detección de desviaciones respecto a las especificaciones de funcionamiento, históricamente se ha hecho hincapié en atacar el origen de dichos errores. Esta sección introduce los intentos de sistematizar y agilizar la programación en si, y en consecuencia acortar la duración de la puesta en marcha y del conjunto del proyecto de automatización.

Se ha introducido previamente a la persona lectora a los DEDS y su fuerte presencia en procesos industriales, y a su modelado a partir de una de estas dos herramientas:

GRAFSET y **PN**. Para la implementación de un **GRAFSET** en un **PLC** se ha solido recurrir a la asignación de una variable booleana a cada etapa [11] [12], evaluándose su estado en cada ciclo de scan del dispositivo. Actualmente los entornos de desarrollo incluyen la herramienta gráfica **SFC**, estandarizada en IEC 61131-3. El enfoque **PN** es más amplio pero su uso es menos generalizado. Por una parte, permite modelar aspectos como concurrencia, sincronización, y recursos compartidos, entre otras ventajas [10]. Hay funcionalidades implícitas en la estructura de una **PN** que requieren de su programación en **GRAFSET**: la gestión de marcado no booleano de lugares mediante contadores asociados a etapas, y el uso de variables internas para funciones de sincronización o recursos compartidos. Por otro lado, la estructura de una **PN** cuenta con una definición matemática formal y los consiguientes métodos de validación, más eficaces que los propuestos para **GRAFSET**, relacionados principalmente con la gestión de la divergencia-convergencia [87].

Numerosos trabajos proponen metodologías de implementación de **PNs** en lenguajes estándar de **PLC** de manera interpretada, es decir, integrando en el propio código la estructura de la red y la gestión de su marcado. El método de [88] para la generación de **LDs** agiliza el desarrollo, la depuración y la reingeniería, además de responder a la necesidad de construir redes de forma sencilla, así como a la evaluación de propiedades y a la simulación de la evolución del marcado. Además de considerar la **PN** como una solución integrada, [89] realiza un estudio sobre los enfoques para su conversión a **LD** en un contexto en el que los sistemas son cada vez más complejos, a lo que añade la necesidad de contar con herramientas de control, análisis, evaluación y simulación de sistemas. El conjunto de reglas definidas por [10] para la implementación de **LDs** supone un avance respecto a desarrollos *ad hoc* meramente intuitivos. La revisión de la literatura llevada a cabo por [90] hace referencia a estas conversiones de **PN** a **LD**, entre otras implementaciones de controladores lógicos.

Se identifican, además, artículos sobre el uso de otros lenguajes de programación, aparte de **LD** [91] [92] [93] [94]. Se trata de listas de instrucciones [95] [96] y propuestas posteriores basadas en distintos lenguajes estándar [97] o **ST** para portabilidad a distintos **PLCs** [98]. En un trabajo reciente, [99] edita tanto estructura como interpretación de una **PN** en un entorno gráfico propio y realiza la conversión automática a código del dispositivo de control.

Las ventajas del soporte de la emulación a lo largo del desarrollo de código de dispositivos de control de sistemas industriales automatizados son similares a las ya descritas en lo que a su **VC** se refiere. El testado de código es continuo a lo largo de su desarrollo, y visual, sin tener que forzar (lo hace el propio modelo virtual) las entradas del **PLC** para observar el comportamiento de las salidas. En aplicaciones de investigación y transferencia se hace especialmente beneficioso en uso de un **DT** para realizar pruebas de partes del programa contra sistemas específicos y no necesariamente disponibles físicamente, así como la reproducción de aquellos escenarios que se requieran.

2.3. Fortalezas y debilidades

Los beneficios previamente comentados del manejo de este tipo de tecnologías en sistemas industriales automatizados se sustentan en la amplia literatura existente:

- Reducción del tiempo de puesta en servicio del sistema de control y el equipamiento asociado [100] [53] [49] [51].



FIGURA 2.2: Emulación de máquina de machos como soporte al proyecto de su reacondicionado.

Fuente: [53].

- Rentabilidad, cuantificada por [59] en un 50 % de reducción de mano de obra en la fase de puesta en marcha al final del proyecto.
- Mayor calidad del resultado obtenido, como se refleja, por ejemplo, en [32].
- La posibilidad de testar y validar sistemas de control basados tanto en PLC [53] y [27] como en sistemas robóticos, como la optimización de trayectorias que presenta [101].
- El soporte tanto a nuevos desarrollos completos como al reacondicionado [53] de maquinaria antigua (Figura 2.2). Se modernizan equipos industriales [102] en respuesta a nuevos requerimientos y cambios tecnológicos, en lugar de dar por concluido su ciclo de vida útil.
- La reconfiguración de sistemas modulares. Autores como [103] ya hablaban de un nuevo paradigma que se manifiesta en la aplicación de modelos modulares para lotes cambiantes de [104], el nuevo marco para VC de [105] y la reconfiguración en un proceso de ensamblado con interacción robot-humano de [106].
- El uso de herramientas de emulación como S3D [53], más adecuadas para el desarrollo de un DT que los simuladores [54].

La literatura se hace eco, asimismo, de las debilidades:

- La complejidad del proceso de modelado [54], lo laborioso del desarrollo de modelos virtuales [58], y los costes económicos asociados [76].
- La falta de personal experto en este ámbito, pese a los esfuerzos por simplificar el modelado [27], así como la escasa implantación de estas herramientas, cada vez más numerosas, en la Pequeña y Mediana Empresa (PyME) e industria automotriz [30]. A pesar de sus numerosos beneficios [57], se percibe escaso interés en la industria [49].
- La emulación contribuye a detectar errores de programación una vez estos se han producido. Son numerosos los intentos no consolidados de implementación sistemática de DEDSs en PLCs documentados en la literatura. De aquí se extrae la oportunidad de investigación descrita en 2.4.1.

- El **DT** es conocido principalmente como herramienta para validar programas de control pero tiene potencial, respaldado por casos de éxito, para emplearse a lo largo de todo el ciclo de vida del sistema industrial automatizado. De aquí se extrae la oportunidad de investigación descrita en 2.4.2.
- El modelado nivel E/S es el de uso más generalizado, suficiente para el testado de código de **PLC** pero que no puede dar soporte a otras tareas. De aquí se extrae la oportunidad de investigación descrita en 2.4.3.

2.4. Líneas de investigación identificadas

De la revisión de la literatura y de las debilidades detectadas se han extraído unas oportunidades de investigación hacia las que orientar este trabajo de tesis. Se entiende que la complejidad del modelado y la escasez de personal competente se mitigará con la implantación creciente de herramientas de digitalización, a lo que se pretende contribuir. Se pone el foco en (i) el apoyo en la sistematización y agilización del desarrollo de software de control, (ii) las aplicaciones en otras fases del ciclo de vida, y (iii) la ampliación del marco de **VC** al ajuste de parámetros operativos, como se detalla en las siguientes secciones.

2.4.1. Implementación sistemática y sencilla de sistemas dinámicos de eventos discretos en PLCs

Autores como [88] demandan la construcción, evaluación y simulación de modelos de **PN** de una manera sencilla. Pese a los numerosos intentos de implementar sistemáticamente **PNs** en **PLCs**, ninguno de ellos puede considerarse efectivo ni de uso generalizado. Es común dicha implementación en un software externo que se comunica con el **PLC**. Las propuestas de hacerlo en el propio dispositivo son dependientes de cómo se programa la lógica de lugares y transiciones para cada una de las aplicaciones, es decir, puede haber tantos métodos como personal técnico encargado de dicha programación, a lo que cabe añadir que cualquier modificación en el código implica su regeneración. Aun habiendo casos de métodos estructurados para dicha lógica, no se ha extendido su implantación:

- Se trata de desarrollos plenamente interpretados, dependientes de la aplicación y del personal técnico. La carga de trabajo relativa al código es considerable y con amplia propensión a errores. Cualquier cambio en estructura, marcado inicial o interpretación de la red afecta al código y obliga a su regeneración. Sirva como ejemplo la herramienta de [99] para la edición de estructura e interpretación de una red.
- No se hace uso de la amplia teoría de validación formal de **PNs**. La automatización y agilización de la conversión de **PN** a **LD** por parte de [107], por ejemplo, se apoya en matrices características pero no son generadas automáticamente ni son analizadas con reglas de validación. Se acostumbra a realizar dicha validación al final del proyecto, a partir del forzado de valores de entrada y la observación del comportamiento de las salidas, bien en el equipamiento ya montado, bien en un **DT** que lo emule.

En consecuencia, la combinación de la representación matricial y los métodos de validación formal puede dar respuesta a la ausencia de metodologías de uso generalizado para la implementación de **DEDSs** en **PLCs**. Si se le añade el potencial de los

DTs para apoyar, testar y validar nuevos desarrollos contra cualquier sistema industrial a través de su emulación, cabe anticipar que se podrá validar dicha metodología con contra procesos muy específicos sin disponer de ellos físicamente.

2.4.2. El gemelo digital más allá de la validación de código de sistemas de control

En un entorno industrial automatizado se dispone de un gran volumen de datos y señales, tanto de sensores y actuadores de la maquinaria como de la interacción con el personal. Utilizarlo para el diagnóstico de la operación puede tener múltiples aplicaciones desde el punto de vista de la formación de personal inexperto o de la evaluación del proceso. Sirvan como ejemplo el [sistema inteligente o Intelligent System \(IS\)](#) de tutorización de [108] y [109], con prestaciones de tutores humanos para la integración de personas con discapacidad en entornos laborales, o el reconocimiento de actividad basado en un sistema de localización WiFi y acelerómetros para la identificación de la postura corporal de [110]. La actividad se interpreta y diagnóstica en función del contexto en el que se produce, pudiendo analizarse el orden de los pasos en la secuencia, la duración de los pasos de los que consta o la duración de la secuencia en su conjunto. La integración, en un marco estándar, de un [PLC](#) o cualquier otro sistema interactivo con conectividad [OPC UA](#), puede hacer posible el diagnóstico contextual automático de un sistema industrial automatizado. Un [DT](#) puede ayudar a su configuración o formar parte de una aplicación exclusivamente virtual, reproduciendo escenarios de interés sin comprometer la producción o la seguridad.

2.4.3. La emulación a nivel componente como soporte a la determinación de parámetros operativos en la fase de puesta en servicio virtual

Aunque tradicionalmente se ha puesto el foco en la puesta en servicio del código del [PLC](#) en la fase final de un proyecto de automatización industrial, tras es mismo tiene lugar el ajuste definitivo de parámetros de variadores de frecuencia que comandan motores asíncronos trifásicos. La línea de trabajo identificada tiene lugar en el ámbito de la eficiencia energética y la sostenibilidad, que es uno de los pilares de la Industria 4.0 [111]. La estimación de consumos en un modelo virtual puede dar soporte a la aplicación de técnicas de optimización, adaptando las velocidades de la instalación a posibles cuellos de botella con el consiguiente ahorro energético, en lugar de que cada mecanismo trabaje al máximo y tener tiempos de espera.

La literatura se hace eco de las virtudes del desarrollo de modelos a nivel componente, que pueden ser utilizados sin necesidad de conocimiento adicional. La existencia de este concepto de modelado abre un nuevo escenario en el que se puede acometer el dimensionado de instalaciones desde la fase de diseño, pudiendo apostar por motores más pequeños, etc. Se tiende al desarrollo de modelos por parte de equipos formados por personal de distintos perfiles, lo cual hace que su posterior uso sea más sencillo.

En términos de sostenibilidad y consumo energético, hay constancia de experiencias previas con modelos virtuales no necesariamente correspondientes a instalaciones reales. Además, se consideran únicamente consumos operativos, faltarían otros escenarios, así como aplicaciones hidráulicas y neumáticas [82]. Asimismo, se plantea

la idoneidad de unificar herramientas de modelado y simulación, y crear plataformas intuitivas. Autores como [26] implementan técnicas avanzadas de optimización a partir de algoritmos genéticos.

Nótese que se ha identificado esta oportunidad de investigación con la tesis ya avanzada, es decir, sin haberla previsto en primera instancia. Se ha estimado de interés acometerla, así como anexas a esta memoria el trabajo preliminar realizado.

2.4.4. Otras oportunidades de investigación

De la revisión de la literatura se extraen otras líneas relacionadas con la emulación aplicada a sistemas industriales automatizados. Pueden resultar de interés para futuros trabajos sin estar directamente vinculadas a esta tesis. Son resumidas en esta sección.

En lo que se refiere al modelado, se identifican oportunidades en aspectos como los que siguen:

- Esfuerzo de modelado. Son numerosas las referencias a la magnitud del esfuerzo que supone el desarrollo de modelos, así como la necesidad de personal competente en la industria.
- La simplificación del modelado. Autores como [27] hacen hincapié en que los fabricantes deberían facilitar modelos, y se debería trabajar en la simplificación del proceso de modelado [32]. En la estandarización se ve una oportunidad para universidades. En términos similares se expresa [50], que ve oportunidades para el desarrollo de conocimientos sobre modelado, operación y control en el ámbito educativo. Mientras, [56] profundiza en la generación automática de modelos, así como en la creación y mantenimiento de librerías, y [112] en la escalabilidad y la modularidad.
- La generación sistemática de DTs. Se percibe interés en la generación automática o semiautomática de modelos, y en la problemática asociada. El método presentado por [28] modela automáticamente el comportamiento de una célula robótica a partir de los programas de robot existentes y de un modelo de simulación de la célula. Los programas de los robots de la planta se cargan en una herramienta de fabricación virtual y, a continuación, se genera un modelo formal a partir de los programas de los robots. A continuación, se calcula automáticamente la lógica de control y se encuentra el orden de ejecución más rápido posible utilizando el modelo generado para formular un problema de optimización. El resultado se analiza continuamente y se valida mediante simulación en la herramienta de fabricación virtual. Mientras, [113] identifica los niveles de detalle y los tipos de modelos adecuados basándose en una percepción jerárquica de las plantas de fabricación automatizadas. El objetivo de su contribución es sistematizar los enfoques de modelización, especialmente la generación automática de modelos de plantas de fabricación. Por último, [56] investiga la posibilidad de generar automáticamente modelos de simulación basados en datos de ingeniería de la planta, utilizables para VC. El mecanismo propuesto es utilizable sin ninguna aplicación adicional específica y puede llevarse a cabo sólo con la herramienta de ingeniería y simulación de la planta. El concepto y el flujo de trabajo se describen en detalle, y el artículo concluye con un estudio de caso basado en una implementación prototípica. Sobre la base

del concepto propuesto, la reutilización de los datos de ingeniería para generar modelos de simulación, y por tanto el uso más amplio de la simulación a lo largo del ciclo de vida, resulta más fácil y por tanto más factible.

En términos de uso de la emulación destacan estas cuestiones:

- Cómo cuantificar el tiempo de desarrollo del modelo virtual y el beneficio de la VC en industria es otro de los aspectos propuestos por [32] y [28].
- Las limitaciones en cuanto a tiempo de respuesta de OPC UA pueden obligar a la experimentación con otros protocolos de comunicación [31] [50].
- Acerca del mantenimiento de modelos con el sistema asociado plenamente operativo destacan trabajos que hacen referencia a la sincronización de los mismos con el mundo real. Las líneas de trabajo abiertas por [63] son relativas a conectividad, estructuración de modelos y datos, y métodos de sincronización. Además, [112] habla de escalabilidad y modularidad, así como de barreras tecnológicas como continuidad digital, sincronización con el mundo real, y la interdisciplinariedad.

En cuanto a aplicaciones de la emulación en el ámbito de interés, cabe destacar lo que sigue:

- Predicción: la consistencia del modelado para la aplicación de técnicas de mantenimiento predictivo es referida por [77]. El mantenimiento proactivo y el uso de sensores modernos para monitorizar un gran número de parámetros son objeto de estudio por parte de [26].
- Métodos formales: cabe destacar su escasa implantación en la industria automotriz [30], y lo necesario de una estructura para aplicarlos en diferentes fases de diseño de un sistema nuevo [28]. La validación de la lógica secuencial comienza antes si se dispone de un modelo formal, creando el sistema de control a partir del mismo.

Por último, y para [63], la conexión en red de toda la información compartida entre socios y que conecta el diseño, la producción y el uso constituye el paradigma presentado de la próxima generación de DTs. De este modo, se salvará la brecha entre la simulación de diseño basada en la física y su uso en las fases de funcionamiento y servicio. El campo de aplicación algo casi ilimitado con mayor concreción e implementación. Las líneas que propone son tres: (i) cómo conectar sistemas de gestión del ciclo de vida del producto, soluciones en la nube y otros dispositivos; (ii) la estructuración de modelos y datos, por ejemplo, mediante el uso de estándares oficiales, así como sus anotaciones semánticas; y (iii) desarrollar métodos y algoritmos para la sincronización de los datos de planta con el DT.

2.5. Definición del trabajo

A partir de las debilidades detectadas y de las oportunidades de investigación identificadas, en primera instancia (Capítulo 4) se plantea experimentar con el DT como herramienta para la VC de sistemas industriales automatizados basados en PLC, en el contexto de la irrupción de tecnologías de la información y la comunicación en los modelos de enseñanza universitarios [114] en virtud del Espacio Europeo de Educación Superior (EEES). Se comprueban los beneficios que aporta en la puesta en servicio de proyectos nuevos y/o de reacondicionado y con objeto de crear un marco

de trabajo para posteriormente dar cobertura a las oportunidades de investigación identificadas.

Posteriormente, el trabajo a realizar se compone de tres tareas principales:

- Según lo descrito en 2.4.1, en lo que se refiere a la agilización de la implementación de DEDSs en PLCs, se ha detectado la ausencia de una metodología común basada en PNs. Si a esto se le suma el potencial de los DTs para apoyar, testar y validar nuevos desarrollos contra cualquier sistema industrial a través de su emulación, se plantea lo siguiente:
 - Por un lado, desarrollar un algoritmo que gestione, mediante bloques de programa fijos, la estructura y la evolución del marcado de una PN presentada matricialmente en memoria de datos y previamente sometida a la amplia teoría de validación de dicho enfoque. De esta manera se codifica únicamente la interpretación de la red, asignando condiciones (consultas acerca de entradas) a transiciones y acciones (escritura de salidas) a lugares. Esta parte manual es la única fuente de potenciales errores en el código.
 - Por otro, definir una metodología novedosa de aplicación sistemática de PNs en dispositivos industriales, independiente de la aplicación y de la plataforma, y probarla contra un proceso en el que afloren las virtudes de una PN sin disponer del mismo, es decir, a partir de su emulación.

Se entiende que de esta manera se pueden desarrollar y poner en servicio proyectos del ámbito de interés de una manera muy eficiente. Las particularidades y contribuciones de este experimento se detallan en el Capítulo 5.

- A partir de lo expuesto en 2.4.2, se propone experimentar con el soporte de la emulación en otras fases del ciclo de vida del sistema industrial automatizado, concretamente en términos de diagnóstico de la actividad. Se diseñará una metodología para implementar un IS mediante cualquier sistema interactivo conectado a través del estándar OPC UA. Un DT del proceso soporta la configuración y la validación, como una aplicación virtual en sí para formación, o antes de la puesta en marcha de un sistema de diagnóstico del sistema real. Se quiere definir un marco para el diagnóstico de la correcta ejecución de tareas automáticas y manuales, la definición de procedimientos basados en la interacción de usuarios expertos con el sistema de control y la formación de personal no cualificado, pudiendo integrarse cualquier dispositivo de control real o emulado y cualquier DT, siempre que tengan conectividad OPC UA. El Capítulo 6 detalla este experimento.
- Finalmente, en línea con lo definido en 2.4.3, y dado que la VC se apoya en el modelado de nivel E/S para el testado de código exclusivamente, se entiende que experimentar con el nivel componente puede permitir someter a prueba otros aspectos clave en la puesta en servicio del sistema de control. Se definirá un marco de trabajo para VC que integre la determinación de parámetros operativos que, respetando los ritmos de trabajo deseados, aseguren el mínimo consumo energético. Más detalles acerca de este trabajo y sus resultados preliminares en el Anexo A.

Capítulo 3

Objetivos, materiales y métodos utilizados

Se da inicio a este capítulo con el enunciado de los objetivos de la tesis y de las hipótesis de trabajo, a partir de la definición del trabajo basada en la revisión de la literatura del Capítulo 2. Posteriormente se pone el foco en el material utilizado y la metodología que ha apoyado esta tesis.

3.1. Objetivos e hipótesis

Esta sección tiene por objeto presentar los objetivos de este trabajo, tanto el principal como los parciales en los que este se descompone, así como las hipótesis de trabajo a las que se quiere dar respuesta la consecución de estos.

3.1.1. Definición de objetivos

Se parte de la definición del objetivo principal de esta tesis, para posteriormente hacer lo propio con los objetivos específicos en los que se divide.

3.1.1.1. Objetivo principal

El objetivo de este trabajo es enunciado como sigue:

Objetivo principal

Evaluación de las tecnologías de digitalización en la agilización del desarrollo y la puesta en servicio de soluciones automatizadas basadas en PLC.

3.1.1.2. Objetivos parciales

En esta sección se enuncia y describe cada **objetivo parcial (OP)** de los cuatro en los que se descompone, para su consecución, el objetivo principal.

Objetivo parcial nº 1

Desarrollo y evaluación de un marco de trabajo, basado en software de emulación de sistemas y procesos industriales, para el testado y la VC del código del sistema de control.

El propósito de este primer **OP** ha sido poner a prueba el cambio de paradigma referido en la literatura en cuanto al papel del **DT** en el testado y validación de sistemas

de control, concretamente del código de PLCs. Se ha trabajado la adquisición de las competencias técnicas necesarias y la generación de material de ayuda para dar respuesta a la problemática que se da en el testado de software en los propios entornos de desarrollo, previamente descrita. Se han comprobado los resultados presentados en los artículos analizados en términos como la agilidad y lo visual de la puesta en servicio, la calidad de la implementación, o la creatividad y motivación del personal. Asimismo, se entiende que este marco de trabajo sirve para dar soporte al resto del trabajo desarrollado en esta tesis.

Objetivo parcial nº 2

Desarrollo y evaluación de una metodología para la implementación de DEDSs en PLCs, a partir de un enfoque basado en PNs, con la gestión de su estructura fija y previamente validada y con el código relativo a la interpretación de la red como única fuente de errores, con el soporte de un DT del proceso a automatizar.

Una vez trabajada la detección de errores en el código del PLC con el apoyo de un DT, el segundo OP pone el foco en el origen de estos. Se ha pretendido simplificar y agilizar la implementación de DEDSs mediante una metodología validada contra procesos clave según la literatura, y no disponibles físicamente. La emulación hace posible dicha validación o VC, y reproduce todos los escenarios que resulten necesarios.

Objetivo parcial nº 3

Desarrollo y evaluación de una metodología para la implementación de una plataforma de diagnóstico de actividad en sistemas industriales automatizados a partir de la integración de cualquier sistema interactivo o PLC con conectividad OPC UA en un IS.

El tercer OP busca experimentar con el DT en otras fases del ciclo de vida del sistema industrial automatizado. Se ha trabajado la integración de un PLC como sistema interactivo, conectado a un DT o no, en un IS ya existente para el diagnóstico contextual de actividad. Se prevén dos posibles aplicaciones. Por un lado, una exclusivamente virtual para la formación de personal inexperto en un sistema emulado. Al no interferir en el sistema físico, se evitan paradas, defectivo y situaciones de riesgo para equipamiento y personas. Por otro, y con el PLC intercambiando señales con el sistema físico, se diagnostica la actividad en el mismo.

Objetivo parcial nº 4

Desarrollo y evaluación de un marco de trabajo, basado en software de emulación de sistemas y procesos industriales, para la estimación de consumos energéticos en receptores y el ajuste, mediante la aplicación de técnicas de optimización, de parámetros operativos en sistemas y procesos industriales.

Este cuarto OP, planteado tras la consecución del primer OP durante el desarrollo de esta tesis, pone el foco en el ajuste de parámetros relativos a velocidades de motores eléctricos posterior a la validación de código de PLC y la puesta en servicio de la solución automatizada. Se acostumbra a trabajar con estos a su velocidad nominal, es decir, los sistemas suelen estar sobredimensionados. Esto conlleva, además del despilfarro energético asociado, la necesidad de cuellos de botella y almacenes

intermedios. El ajuste manual de velocidades en sistemas equipados con variadores de frecuencia puede ser motivo de extensión de plazos de entrega y de generación de defectivo. Un **DT** puede estimar el consumo de receptores y apoyar el uso de técnicas de optimización en la determinación de parámetros operativos en sistemas industriales automatizados, para los tiempos de ciclo y los ratios de producción requeridos.

3.1.2. Definición de hipótesis

A los objetivos planteados se les pretende dar respuesta mediante las siguientes hipótesis de trabajo asociadas a cada uno de ellos.

Hipótesis relativas al objetivo n° 1

Hipótesis 1.1: *El uso de un **DT** de un sistema industrial automatizado basado en **PLC** reduce la duración de la fase de puesta en servicio de su código y permite obtener resultados de mayor calidad, con menor disponibilidad de equipamiento y menor riesgo para el mismo y para las personas.*

Hipótesis relativas al objetivo n° 2

Hipótesis 2.1: *La definición matemática formal de una **PN** y la amplia teoría de validación estructural de la misma hacen posible implementar **DEDSs** en **PLCs** utilizando bloques de programa mayoritariamente fijos, independientes de la aplicación y del dispositivo, y sin consumir más recursos de CPU que otros enfoques similares.*

Hipótesis 2.2: *El uso de un **DT** hace posible validar nuevos desarrollos contra sistemas no disponibles físicamente de una manera visual y ágil.*

Hipótesis relativas al objetivo n° 3

Hipótesis 3.1: *La integración, a modo de sistema interactivo y través de **OPC UA** en un **IS**, de un **PLC** que controla un sistema industrial automatizado emulado a través de su **DT**, hace posible el diagnóstico de actividad en aplicaciones virtuales en un contexto de entrenamiento/formación de personal inexperto.*

Hipótesis 3.2: *La integración, a modo de sistema interactivo y través de **OPC UA** en un **IS**, de un **PLC** que controla un sistema industrial automatizado, hace posible el diagnóstico de actividad en aplicaciones industriales reales.*

Hipótesis relativas al objetivo n° 4

Hipótesis 4.1: *La estimación de consumos energéticos en modelos virtuales de nivel componente posibilita el ajuste de parámetros operativos a la máxima eficiencia para los ratios productivos previstos en sistemas industriales automatizados dotados de control de velocidad.*

3.2. Materiales

Para el desarrollo de esta tesis se ha hecho uso de una serie de recursos. Se trata de controladores industriales y sus entornos de desarrollo, herramientas de emulación

para el desarrollo y uso de **DTs**, y otras herramientas específicas para aplicaciones concretas. Todo esto se detalla en cada uno de los experimentos realizados, en términos de fabricantes, referencias, versiones y similares.

3.3. Metodología

Para el desarrollo de esta tesis doctoral se ha planteado el uso de diferentes metodologías que se relacionan de forma directa con cada uno de los objetivos de la misma, ya que para la consecución de cada uno de ellos no necesariamente se ha procedido de la misma manera. Se ha adaptado, en consecuencia, cada metodología a cada trabajo a realizar, con objeto de obtener mejores resultados.

Por una parte, se ha llevado a cabo la gestión del proyecto en colaboración entre el doctorando y el director, siendo el doctorando el responsable último del cumplimiento de los objetivos en los plazos establecidos, bajo la supervisión del director, con el que ha celebrado reuniones periódicas para la revisión de la situación del proyecto y la adecuación de la metodología planteada en cada hito definido.

La metodología de investigación seguida en esta tesis se basa en estrategias de diseño, creación, experimentación y la validación empírica de los resultados obtenidos. Se ha empezado por una revisión de la literatura sobre la temática de interés, analizándose tanto trabajos de referencia como publicaciones recientes. Posteriormente se han llevado a cabo las acciones que se describen de cara a la consecución de los objetivos parciales propuestos.

Para el **OP1**, *Desarrollo y evaluación de un marco de trabajo, basado en software de emulación de sistemas y procesos industriales, para el testado y la VC del código del sistema de control*, se ha llevado a cabo un análisis de herramientas de emulación, la preparación de modelos virtuales y material de apoyo, y la posterior realización de actividades soportadas por **DTs** en el ámbito académico. Conocidas las limitaciones del procedimiento convencional de desarrollo de proyectos de automatización industrial, se han diseñado y desarrollado proyectos de automatización aplicando un procedimiento innovador basado en la **VC** del sistema de control, extrayendo resultados cualitativos en la comparativa entre ambos métodos de trabajo.

En cuanto al **OP2**, *Desarrollo y evaluación de una metodología para la implementación de **DEDSs** en **PLCs**, a partir de un enfoque basado en **PNs**, con la gestión de su estructura fija y previamente validada y con el código relativo a la interpretación de la red como única fuente de errores, con el soporte de un **DT** del proceso a automatizar*, se ha definido una metodología para la implementación de **DEDSs** en **PLCs** a través de un algoritmo que gestiona la evolución del marcado de una **PN**. De cara a obtener una forma sistemática de programar este tipo de sistemas, independiente tanto de la aplicación concreta como del dispositivo de control utilizado, en primera instancia se ha diseñado el algoritmo, fijo en mayor parte y que utiliza la estructura de la red en forma matricial previamente validada. Posteriormente se han especificado los pasos para su aplicación, y finalmente se han definido casos contra los que poner a prueba ambos diseños.

En lo que respecta al **OP3**, *Desarrollo y evaluación de una metodología para la implementación de una plataforma de diagnóstico de actividad en sistemas industriales automatizados a partir de la integración de cualquier sistema interactivo o **PLC** con conectividad **OPC UA** en un **IS***, se ha diseñado una herramienta de diagnóstico automático del contexto

de un sistema industrial automatizado. Se basa en la integración de un sistema interactivo (PLC) en un marco genérico a través del estándar de comunicaciones OPC UA. Se disponía previamente de un metamodelo y su núcleo de ejecución. Se han caracterizado distintos procedimientos a verificar en casos objeto de estudio, tanto con DT, para aplicaciones virtuales de formación de personal inexperto, como con sistema físico.

Finalmente, y en lo referente al OP4, *Desarrollo y evaluación de un marco de trabajo, basado en software de emulación de sistemas y procesos industriales, para la estimación de consumos energéticos en receptores y el ajuste, mediante la aplicación de técnicas de optimización, de parámetros operativos en sistemas y procesos industriales*, se ha dotado a modelos nivel componente de la estimación del consumo energético de receptores de un sistema industrial automatizado. Con objeto de evitar cuellos de botella y almacenes intermedios como consecuencia del funcionamiento de motores a velocidad nominal, se han diseñado experimentos en los que se ha determinado el gasto a distintos ritmos de trabajo, para elegir los parámetros operativos idóneos para los ratios productivos previstos.

Los experimentos han sido desarrollados por el doctorando y Personal Docente e Investigador (PDI) tanto en I+T como en actividad reglada, según se especifica en la Parte II del presente documento. Se han ejecutado ciclos de funcionamiento en PLC de cada sistema industrial automatizado objeto de estudio, con casos de carácter automático, semi-automático, o con interacción con personas, según el caso.

Los resultados de estas actividades han sido extraídos y analizados con dos enfoques: (i) el cualitativo, basado en la observación del funcionamiento y en testimonios de participantes en la actividad, y (ii) el cuantitativo, en los experimentos que más se han adecuado a ello, con mediciones de período de scan de PLCs con el fin de comparar el consumo de recursos, estimaciones de consumos energéticos en un DT, etc.

Por último se ha realizado la divulgación de desarrollos y aportaciones a la comunidad científica, en forma de artículos para conferencias internacionales y para revista.

Parte II

Desarrollo de la tesis

Capítulo 4

Introducción al desarrollo de la tesis

La aplicación de tecnologías de digitalización para la emulación de sistemas industriales abre un nuevo paradigma en el desarrollo de proyectos de soluciones automatizadas. El testado del software de dispositivos de control se realiza de una manera visual antes de disponer del equipamiento montado y/o operativo, haciendo posible la validación virtual del código. La puesta en servicio del sistema es más ágil en un proyecto apoyado por un **DT** que en un proyecto convencional. Asimismo, la emulación puede tener aplicación más allá del soporte al desarrollo de código, pudiendo tener presencia a lo largo de todo el ciclo de vida del producto.

En esta tesis se analiza la aplicación de **DTs** en implementaciones del ámbito de la automatización industrial. Se estudian distintas problemáticas que se producen tanto en el ámbito académico como en actividades de I+T. Por una parte, se realiza la puesta en servicio virtual de aplicaciones que posteriormente se ejecutan en el sistema real, en términos de testado de código, determinación de parámetros operativos o validación de una herramienta de diagnóstico de actividad. Por otra, se exploran aplicaciones exclusivamente virtuales como la validación de un algoritmo contra un proceso específico emulado en el que poner a prueba la validez del desarrollo, o el diseño de una plataforma de entrenamiento de personal inexperto.

Sirva el diagrama de la Figura 4.1 como introducción a los Capítulos 5, 6 y 7, así como al Anexo A. En la misma se sintetizan los experimentos asociados a cada **OP** de este trabajo, desglosándose la información como sigue:

- **Capítulo** o anexo en el que se detalla el experimento realizado.
- **Objetivo parcial** con el que se alinea.
- **Descripción** o enunciado del propio objetivo parcial.
- **Contribución** llevada a cabo, identificándose los artículos de conferencia con la letra *C* y los de revista con la letra *J*.
- **Fase(s) del ciclo de vida del sistema automatizado** (desarrollo del sistema de control, su puesta en servicio virtual o real, y/o fase operativa) en la(s) que interviene o tiene influencia la contribución, y en qué términos.

Capítulo	Objetivo parcial	Descripción	Contribución	Fase del ciclo de vida del sistema automatizado en la que interviene o tiene influencia la contribución		
				Desarrollo	VC	Operativa
5	[OP1]	Desarrollo y evaluación de un marco de trabajo basado en software de emulación de sistemas y procesos industriales para el testado y VC del código del sistema de control.	[C1] [C2]	Testado del sistema de control (código de PLC) contra la emulación del sistema físico.	Validación del sistema de control (código de PLC) contra la emulación del sistema físico.	Puesta en servicio del sistema de control (código de PLC) en el sistema físico.
6	[OP2]	Desarrollo y evaluación de una metodología para la implementación de DEDSs en PLCs, a partir de un enfoque basado en PNs, con la gestión de su estructura fija y previamente validada y con el código relativo a la interpretación de la red como única fuente de errores, con el soporte de un DT del proceso a automatizar.	[C3][J1]	Testado del sistema de control (nuevas implementaciones sistemáticas de código de PLC) contra el sistema emulado.	Validación del sistema de control (nuevas implementaciones sistemáticas de código de PLC) contra el sistema emulado.	
7	[OP3]	Desarrollo y evaluación de una metodología para la implementación de una plataforma de diagnóstico de actividad en sistemas industriales automatizados a partir de la integración de cualquier sistema interactivo o PLC con conectividad OPC UA en un IS.	[C4]	Testado del IS (definición de procedimientos y diagnóstico de actividad) contra la emulación del sistema físico.	Validación del IS contra la emulación del sistema físico.	Puesta en servicio de aplicaciones de diagnóstico en el sistema físico.
Anexo A	[OP4]	Desarrollo y evaluación de un marco de trabajo basado en software de emulación de sistemas y procesos industriales para la estimación de consumos energéticos en receptores y el ajuste de parámetros operativos mediante técnicas de optimización.	En desarrollo	Testado del sistema de control (consignas de velocidad a VSDs) contra la emulación del sistema físico.	Validación del sistema de control (consignas de velocidad a VSDs) contra la emulación del sistema físico.	Puesta en servicio del sistema de control (consignas de velocidad a VSDs) en el sistema físico.

FIGURA 4.1: Estructura de la tesis.

En los capítulos mencionados, y que vienen a continuación, se describen los experimentos siguiendo esta estructura:

- **Introducción.**

Se presenta el experimento realizado en base a tres aspectos: el contexto en el que se ha llevado a cabo, la definición del problema a responder y la justificación del uso de un DT.

- **Evaluación empírica.**

- **Pregunta(s) de investigación.**

Se enuncia/enuncian la **pregunta de investigación o Research Question (RQ)**/las **RQs** a la/las que se quiere dar respuesta.

- **Caracterización del ensayo.**

- **Material utilizado.**

Se listan los recursos materiales y las versiones específicas de software que se han requerido, es decir, se facilita la información necesaria para la reproducción del ensayo.

- **Caracterización del DT.**

Se expone cómo se ha diseñado el proceso emulado, así como su conectividad y la configuración de la que se ha hecho uso.

- **Trabajo realizado.**

Se concretan las condiciones en las que se han llevado a cabo los ensayos.

- **Metodología.**

Se describen los pasos de la metodología seguida.

- **Cierre del experimento.**

Se determina cómo se han obtenido los resultados.

- **Resultados.**

Se presentan los mismos en tablas y se resaltan los más destacados, identificando dónde se aprecian diferencias.

- **Discusión.**

Se resumen los resultados, se determina lo que dicen respecto a cada **RQ** y se valora el posible cumplimiento de aquello que se perseguía con el experimento.

- **Trabajo relacionado.**

Se da a conocer lo que ha supuesto esta aportación, en términos de generación de actividad relacionada.

- **Conclusión y trabajo futuro.**

Se resume el experimento, haciendo referencia a aquello que se haya diseñado; se resaltan los resultados obtenidos y la aportación de la aplicación de tecnologías de digitalización; y se identifican posibles líneas de trabajo futuro.

Capítulo 5

Marco para testado y puesta en servicio virtual de programas de control de sistemas industriales automatizados.

Fabricantes de maquinaria e integradores de sistemas afrontan una demanda creciente en términos de reducción de plazos de entrega de nuevas aplicaciones o reacondicionado de soluciones automatizadas antiguas [59]. En un entorno de gran complejidad, la puesta en servicio de sistemas automatizados debe ser rápida, lo cual trae consigo la necesidad de agilizar el testado y la validación del sistema de control, uno de las fases críticas en un proyecto.

La capacidad computacional del nuevo equipamiento informático y la irrupción de herramientas software para la emulación de maquinaria y procesos industriales sitúan al personal técnico frente a un nuevo paradigma. Se evalúa el código de dispositivos de control antes de la fase de puesta en servicio, lo cual es conocido como **VC** [27], pero no hay suficientes profesionales competentes en estas tecnologías [58].

El académico es uno de los entornos donde se desarrollan los profesionales del futuro. El uso de **DTs** en este ámbito hace posible una mejor gestión de recursos materiales, evitando la manipulación de sistemas reales y costosos y reduciendo el ratio de ocupación de laboratorios físicos, aumentando su disponibilidad. El testado de programas de **PLC** en su entorno de desarrollo requiere un buen conocimiento del proceso a automatizar, así como una correcta reproducción de su funcionamiento, asignando valores apropiados a variables y siguiendo las secuencias deseadas. La emulación ayuda a entender el sistema o proceso, y reproduce dichas señales, apoyando al alumnado en el desarrollo de programas de mayor calidad, la adquisición de competencias técnicas y transversales, y los consiguientes resultados académicos.

Este capítulo tiene por objeto detallar un primer experimento con **DTs** aplicados a sistemas industriales automatizados. Mediante el mismo, el cual ha tenido lugar en clases regladas en laboratorios de la Universidad, se ha desarrollado un marco de trabajo con estas tecnologías y se han adquirido las habilidades necesarias para posteriores trabajos.

5.1. Introducción

En esta sección se introduce al lector al experimento realizado en el contexto de la impartición de contenidos de robótica y automatización industrial a alumnado universitario, concretamente en unas sesiones de carácter práctico en un laboratorio dotado de una célula robótica a modo de estación de trabajo para identificación, control de calidad y clasificación de carcasas de dron. Los participantes han tenido por objetivo la implementación del control, comparando el procedimiento convencional y el apoyado por un DT.

5.1.1. Automatización de sistemas industriales

Gran parte de los sistemas industriales a automatizar son DEDSs: sistemas de fabricación flexibles y/o discretos dotados de robots y/o máquinas programables con evolución concurrente, sincronización y/o recursos compartidos. Un DEDS es un sistema dinámico y asíncrono, donde las transiciones de estado se inician por eventos que ocurren en instantes discretos de tiempo [5]. El PLC es el dispositivo de control de uso más generalizado en entorno industrial. Orientado a sistemas del tipo descrito, se basa en la información del proceso y del operador [3]. Se diferencia de un ordenador en los términos siguientes [4]: (i) ejecución cíclica del programa; (ii) fiabilidad; (iii) adaptabilidad a la presencia de ruido eléctrico, vibraciones, temperaturas y humedad extremas; y (iv) fácil mantenimiento.

5.1.1.1. Testado y validación convencional de sistemas de control

La Figura 1.8 muestra, en el diagrama identificado como A, las fases de un proyecto convencional de automatización industrial: diseño; desarrollo del sistema de control; instalación; y puesta en servicio del sistema de control. Se acostumbra a testar el código de PLC a lo largo de su desarrollo y a partir del forzado entradas y/o variables, y la visualización de salidas y/o variables en el propio entorno del fabricante del dispositivo, sin visión directa del comportamiento del proceso. La validación de una solución automatizada tiene lugar durante la fase de puesta en servicio de la misma. Se hace uso del sistema real, incluso con el equipamiento ya montado, bien en las instalaciones de la empresa desarrolladora, bien en las de la compañía cliente.

Trabajar exclusivamente en el entorno de desarrollo del controlador, sin un procedimiento de test riguroso y un buen conocimiento del proceso a automatizar puede hacer que el sistema de control llegue incompleto o con errores a la puesta en servicio. Dicha fase final del proyecto puede tener una duración considerable, con gastos inesperados y/o innecesarios, un coste personal y social para el personal técnico a cargo, y posibles daños/lesiones en equipamiento/personas. La consiguiente pérdida de posicionamiento del fabricante en el mercado, al ser factores clave el tiempo de maquinaria parada en proyectos de reacondicionado y el plazo de entrega en nuevos desarrollos completos, puede afectar a su reputación y condicionar pedidos futuros. En definitiva, la puesta en servicio convencional no garantiza plazos de entrega cortos ni su cumplimiento.

5.1.1.2. El gemelo digital en el testado y validación de sistemas de control

La irrupción del DT ha traído consigo el concepto de la VC de sistemas industriales automatizados. A continuación se repasan estos conceptos y aspectos relativos a la conectividad entre sistema emulado en un PC con el sistema de control.

5.1.1.2.1 Herramientas de emulación de sistemas industriales

Un **DT**, una de las tecnologías de digitalización con mayor potencial en numerosos campos en la industria [25], es un equivalente virtual e informatizado que imita el comportamiento de un sistema físico para realizar el mismo trabajo y producir los mismos resultados. Se replica el mundo real con suficiente concreción para que el dispositivo de control no pueda distinguirlo del mismo, de manera que puede ser utilizado para probar la funcionalidad o la lógica de un **PLC**, o un programa de robot.

5.1.1.2.2 Puesta en servicio virtual

La Figura 1.8 muestra, en el diagrama identificado como *B*, las fases de un proyecto apoyado por un **DT**. El periodo de testado y validación virtual puede ir desde la fase de diseño a la de puesta en servicio. Un **DT** hace posible someter a pruebas los sistemas de fabricación y los programas de control asociados de manera virtual y previamente al montaje de la instalación real [27], lo que es conocido como **VC**. Se posibilita la realización de pruebas sin originar paradas y el testado de situaciones que en un sistema real puede resultar caras o complejas de reproducir [28], tanto en **PLCs** como en robots y sistemas de visión artificial.

Son numerosos los beneficios, tratados en la revisión de la literatura del Capítulo 2, del soporte de un **DT**. Destaca la menor duración de la puesta en marcha del sistema industrial automatizado. Se detectan comportamientos anómalos con antelación, y se reproducen situaciones que en el sistema físico serían costosas o complejas [28], u ocasionar paradas en la producción.

5.1.1.2.3 Conectividad entre sistema de control y gemelo digital

OPC UA se ha convertido en el estándar para la comunicación entre los sistemas de control industrial y el software de terceros, como por ejemplo las herramientas de emulación. Basado en el principio cliente-servidor, es independiente del proveedor, es decir, permite conectar **PC** y **PLC** de cualquier fabricante [31]. Esto hace posible probar los sistemas de control mediante un **DT** del proceso, leyéndose y escribiéndose de forma transparente variables del dispositivo.

Los **PLCs** de última generación incluyen conectividad **OPC UA**, ya que este es considerado el estándar de la Industria 4.0. Existe software externo para controladores que no proporcionan esta característica.

Para la conexión entre planta virtual y sistema de control puede seguirse una de las configuraciones [32], ya mostradas en la Figura 1.9:

- Planta virtual y controlador virtual, o **SIL**: consiste en combinar un modelo preciso de la planta a controlar con un **PLC** emulado. Es una alternativa relativamente barata, pero al no utilizar equipamiento, descuida aspectos relativos a la seguridad como colisiones de robots o daños personales.
- Planta virtual y controlador real, o **HIL**: se utiliza todo el hardware posible, como **PLC**, módulos auxiliares y dispositivos de campo. Una de sus ventajas es que se crea un entorno idéntico o muy similar al de la instalación real y el código se testa directamente en el dispositivo de control. Por el contrario, esta configuración supone un coste más elevado, menos flexible ante cambios durante la fase de diseño. Además, el riesgo de accidentes y lesiones es mayor.

5.1.2. Definición del problema

La problemática de partida de este experimento es habitual en el desarrollo de cualquier solución automatizada en entorno industrial o en su reproducción en el ámbito académico: la escasa o nula disponibilidad del equipamiento a la hora de validar el código de control. En este caso concreto, la existencia de una única unidad del proceso a automatizar.

Se ha solido organizar el acceso al equipamiento en turnos de duración limitada, con las consiguientes dificultades para que el alumnado conozca la célula de manufactura y entienda el funcionamiento deseado. Además, el hecho de verificar previamente el código desarrollado a partir del forzado de entradas del dispositivo de control en su propio entorno de desarrollo no garantiza alcanzar soluciones automatizadas que den respuesta a las especificaciones. Por ello, la puesta en servicio del sistema acostumbra a ser larga, y el equipamiento sufre desgaste por su sobreutilización.

A partir de la revisión de la literatura, que se hace eco del uso de tecnologías de digitalización ante la escasa disponibilidad de laboratorios físicos [115], se ha considerado que el uso de la emulación puede servir para dar respuesta a la problemática descrita, haciendo posible trabajar con varias maquetas en paralelo, en forma de DT. Esto puede traer consigo una mejor comprensión del proceso a automatizar y del trabajo a desarrollar, así como un testado más visual del código. Se agiliza la implementación de la solución, acortándose el plazo de entrega y requiriendo un uso mínimo del equipamiento físico. El alumnado obtiene mejores resultados, tanto en términos de calidad del trabajo realizado, como de rendimiento académico, además de cubrir, a futuro, la demanda de personal técnico competente en estas tecnologías.

5.2. Evaluación empírica

Para la evaluación de este experimento se ha procedido como se detalla a continuación. Tras enunciar la pregunta de investigación, se ha caracterizado el ensayo a realizar y se han obtenido y discutido los resultados obtenidos.

5.2.1. Pregunta de investigación

A partir del contexto descrito y de lo extraído de la literatura, se ha enunciado una RQ:

RQ01

¿Posibilita un DT una puesta en servicio ágil de código de PLC, con resultados de calidad y menor uso de equipamiento y riesgos asociados?

5.2.2. Caracterización del ensayo

Esta sección está compuesta de cinco bloques. En el primero de ellos se enumera el material utilizado. Posteriormente se detallan las características del sistema emulado en el experimento. Se sigue detallando el trabajo realizado y las condiciones en las que se ha llevado a cabo, así como la metodología seguida. Finalmente se cierra el ensayo con la obtención de resultados.

5.2.2.1. Material utilizado

Los recursos requeridos, los cuales no han sido necesariamente utilizados en su totalidad por todos los equipos de trabajo constituidos según se detalla en la descripción de la metodología seguida, han sido los siguientes. Se especifican, si procede, las versiones de software de las que se ha dispuesto:

- La célula robótica objeto de estudio (Figura 5.1) consta de las siguientes partes para la identificación, control de calidad y clasificación de carcasas de dron (Figura 5.2):
 - Una base equipada con un robot KUKA Agilent para las operaciones de manipulación. Podría tratarse de un pórtico con control coordinado de ejes X, Y y Z sin control de orientación.
 - Una mesa lineal, controlada mediante un servovariador, para el movimiento de piezas a través de cuatro posiciones: entrada, medición de altura para determinar si la pieza es buena o mala, identificación del color mediante un sensor cromático, y salida para que el robot transporte la pieza a la zona que le corresponda.
 - Un sistema de seguridad basado en un vallado perimetral y un área de acceso protegida por una barrera, para que el personal cargue/descargue el material sin peligro.
- PLCs con conectividad [OPC UA](#).

Se han utilizado dispositivos Siemens 1516F-3 PN/DP.
- Panel [HMI](#) para la interacción con usuarios.

Concretamente un panel táctil de la serie TP700 Comfort de Siemens.
- Herramientas software:
 - Software de simulación y programación offline de robots.

En este caso, RobotStudio 2019.1 de ABB.
 - Entorno de desarrollo para dispositivos de control.

Se ha utilizado TIA Portal V15.1 de Siemens, así como PLCSIM Advanced V2.0 SP1 para emular [PLCs](#).
 - Un sistema [SCADA](#) para la interacción con usuarios.

Se ha dispuesto de una aplicación desarrollada en Wonderware InTouch 2014 R2.
 - Una plataforma para dotar de conectividad [OPC UA](#) al [SCADA](#).

Se trata de KEPServerEX 6.2 con Wonderware OPC UA Client Service.
 - La versión emulada del sistema de manufactura, en forma de [DT](#) en [S3D](#).

5.2.2.2. Caracterización del modelo virtual

En esta sección se introduce al lector a las características con las que cuenta el [DT](#) con el que se ha experimentado.

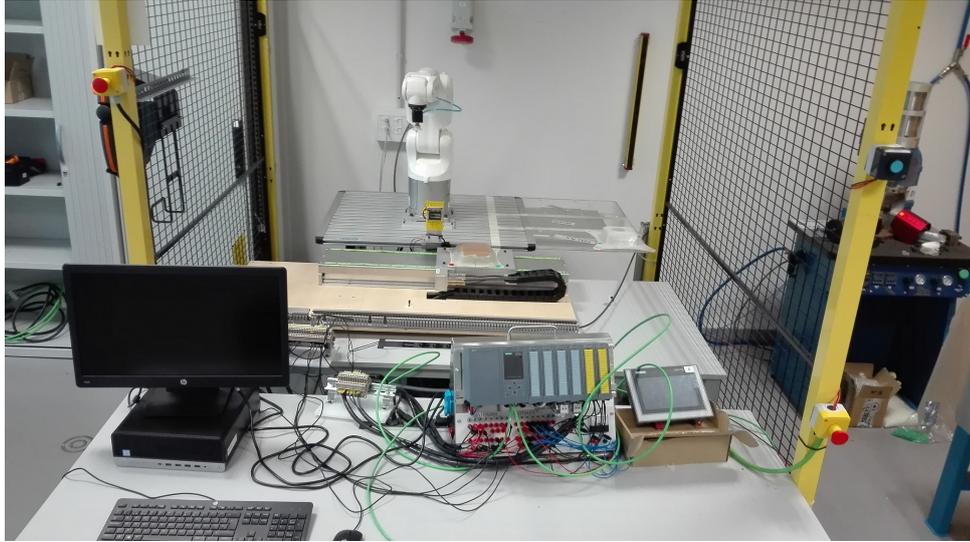


FIGURA 5.1: Célula de manufactura dotada de mesa lineal y robot.

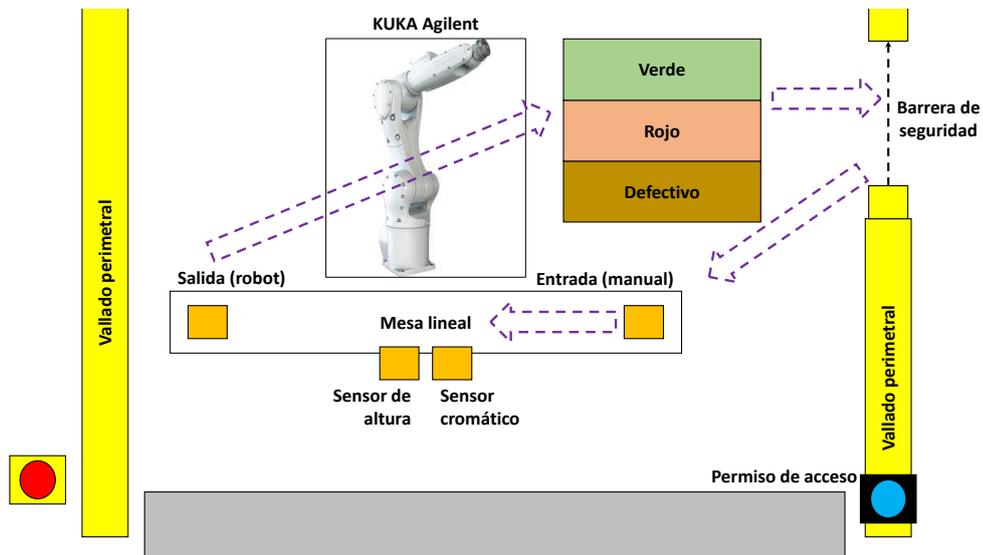


FIGURA 5.2: Diagrama explicativo de la célula de manufactura.

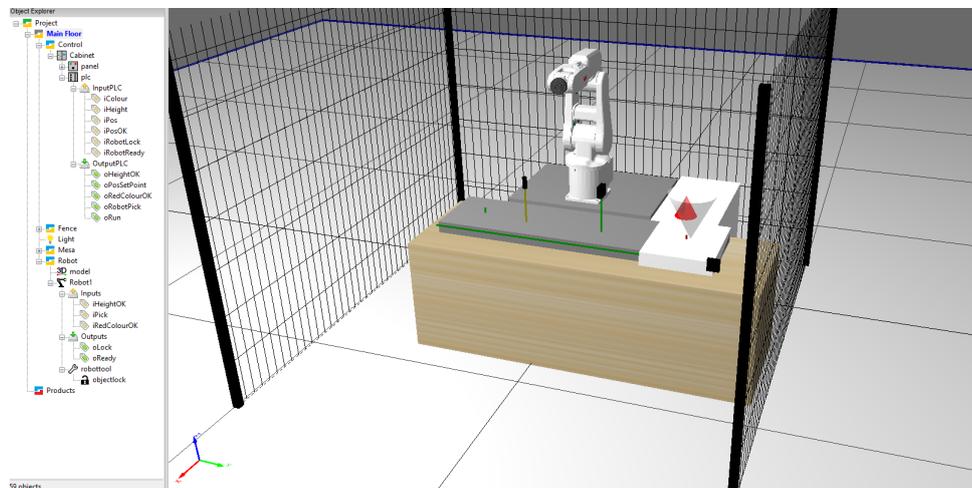


FIGURA 5.3: Gemelo digital de la célula de manufactura.

5.2.2.2.1 Sistema objeto de estudio

Se ha emulado la célula de manufactura descrita.

5.2.2.2.2 Diseño del modelo virtual

Se ha utilizado [S3D](#), de licencia gratuita y en cuyo manejo se cuenta con la experiencia necesaria. El diseño del [DT](#) ha tenido por responsable a personal docente para apoyar la [VC](#) de la célula, ya que la actividad académica en la que ha tenido lugar este experimento no ha tenido por objetivo el modelado sino el desarrollo de competencias relativas a la programación de dispositivos de control y la puesta en servicio de sistemas automatizados.

5.2.2.2.3 Particularidades del modelo virtual

El [DT](#) es el de la Figura [5.3](#). Incluye la mesa lineal, el servovariador, los sensores y el robot, así como el cableado virtual del sistema, como el de sensores/(pre)actuadores con entradas/salidas del objeto [PLC](#), y salidas de este a entradas del objeto robot. Este último, conectado a su controlador, reproduce los movimientos programados en el mismo.

5.2.2.2.4 Conectividad

Se ha utilizado el estándar [OPC UA](#) para comunicar sistema de control y [DT](#).

TABLA 5.1: Señales de control en el modelo virtual.

Salidas robot		Entradas PLC		Entradas robot		Salidas PLC
oLock	⇒	iRobotLock		iPick	⇐	oRobotPick
oReady	⇒	iRobotReady		iHeightOK	⇐	oHeightOK
				iRedColourOK	⇐	oRedColourOK

La Tabla [5.1](#) muestra las señales utilizadas en el [DT](#). Nótese la correspondencia entre variables de ambos dispositivos de control. Con el propósito de coordinar las tareas a realizar se han conectado estas señales entre sí en el modelo virtual, de la misma manera en la que se procede con sensores a entradas del [PLC](#) y salidas a actuadores.

Para la conexión de señales de los controladores con el [DT](#) a través de [OPC UA](#) se han tenido en cuenta varios aspectos. En el caso del [PLC](#), los siguientes:

- Se ha habilitado el servidor [OPC UA](#) de la CPU en una dirección IP y un puerto.
- Se ha asignado a todas las variables compartidas el mismo nombre que en el emulador.
- Se ha desactivado la actualización automática de memorias asociadas a entradas, para su escritura por parte del modelo virtual.

En lo que respecta al robot:

- Se ha asignado a todas las variables compartidas el mismo nombre que en el emulador.

- Al no disponer el robot de un servidor **OPC UA**, se ha utilizado el software **S3D OPC UA Server v0.1.6** para tal fin.
- Una vez agregado el robot en dicha herramienta, se ha especificado el modelo de dicho dispositivo en una lista desplegable y se han incorporado las variables a utilizar en **S3D**.

Por último, y para el **DT**:

- En el objeto **PLC** del **DT** se hace referencia a la citada dirección IP y puerto del servidor **OPC UA**.

5.2.2.2.5 Configuración

Se ha dispuesto el sistema de control según una configuración **SIL** y/o **HIL**, según la dinámica de trabajo seguida por cada grupo de personas participantes.

5.2.2.3. Trabajo realizado

El trabajo realizado ha consistido en la programación, por parte de alumnado del Máster Universitario en Ingeniería Industrial, del **PLC** en respuesta a las especificaciones de funcionamiento coordinado con el robot, de cuyo programa ya se disponía. Para una configuración **HIL** se ha recurrido al dispositivo físico, y para **SIL** se ha emulado el **PLC** en **PLCSIM Advanced**. Personal y dispositivos de control han interactuado a través de dos interfaces ya operativas: una aplicación **SCADA** y un **HMI**.

El ciclo de trabajo deseado consiste en la siguiente secuencia. Se acciona el botón de permiso para el desbloqueo de la barrera y el consiguiente acceso del operario al recinto, de cara a depositar una carcasa de dron en el puesto de entrada. El controlador suministra en todo momento las señales adecuadas para que la mase lineal traslade la pieza de cada punto al siguiente, y lee las señales procedentes de los sensores. Se da por finalizado el ciclo cuando la carcasa, previamente medida y clasificada, es evacuada por el robot, y el carro retrocede a la posición inicial. Simultáneamente, el control realiza el contaje de todas las piezas manipuladas, así como el de aquellas de dimensiones fuera de rango y el tiempo total en situación de parada. En cada uno de los tres casos se dispone de un pulsador para su puesta a cero.

Para cumplir con el funcionamiento requerido, se ha propuesto esta estructura básica para el programa del **PLC**:

- Ejecución del bloque de arranque: puesta a cero de salidas y variables internas, y activación del servovariador.
- Ejecución continua de estos bloques de programa en el ciclo de scan del dispositivo.
 - La secuencia de trabajo descrita, la cual puede ser de utilidad para el aprendizaje de la metodología **GRAFCET** y la programación de estos diagramas en **SFC**.
 - Con la carcasa en el puesto de control de calidad, calcular su altura a partir de la señal de entrada analógica asociada. Comparación y determinación de si la altura se encuentra dentro del rango adecuado.

- Cálculo de la señal de consigna analógica de posición a suministrar al servovariador, a partir de lo que se determine en la secuencia principal: 0 %, 40 %, 60 % ó 100 %.
- Otras funciones.

5.2.2.4. Metodología

En el contexto de un proyecto semestral e interdisciplinar se han trabajado aspectos de distintas materias, tales como el diseño y manufactura de un prototipo de carcasa de dron, el control de la mesa lineal de la célula robótica, y la referida implementación del sistema de control.

Dada la disponibilidad en laboratorios de MU-GEP de una unidad del sistema descrito, durante los años académicos anteriores se han detectado estas limitaciones:

- Entender tanto el proceso a automatizar como el funcionamiento deseado es complicado.
- La depuración de código de PLC en su propio entorno de desarrollo no garantiza el desarrollo de programas de calidad, ya que no es una herramienta intuitiva para que el alumnado reproduzca todos los escenarios que se puedan dar.
- Acceso muy reducido al equipamiento, a partir un calendario de turnos.
- Puestas en servicio que se alargan en el tiempo, como consecuencia de la existencia de errores en el código.

En el año académico en el que ha tenido lugar este experimento se ha incorporado al material disponible para el alumnado, dividido en grupos de tres integrantes, un DT del sistema de manufactura. Se ha implementado el sistema de control siguiendo tres formas de proceder distintas, asignando una de ellas a cada grupo. En algunos casos se ha dispuesto del apoyo de un modelo virtual del sistema solicitándose a todos los grupos participantes el cumplimiento de idénticas especificaciones de funcionamiento (ver Tabla 5.2 y Figura 5.4). Se han analizado los beneficios de la VC a partir de las condiciones de trabajo propias y la observación de las ajenas:

TABLA 5.2: Caracterización del experimento: asignación de condiciones de trabajo.

Identificador	a	b	c
Nº de grupos	3	4	6
Sistema de control	Emulado + real	Emulado + real	Emulado
Proceso	Real	Emulado (DT) + real	Emulado (DT)
Puesta en servicio	Real	VC + real	VC
A quiénes observan	a, b, c	a, b, c	a, b, c

- (a) Tres grupos sin DT y realizando exclusivamente la puesta en servicio del sistema físico, es decir, se ha seguido el procedimiento convencional, como en cursos anteriores. Se han utilizado PLCs reales y se ha compartido la maqueta según un calendario para el uso coordinado de la misma. Se ha partido del desarrollo del software de control, testándolo en su propio entorno de desarrollo, y se ha procedido a su puesta en marcha.

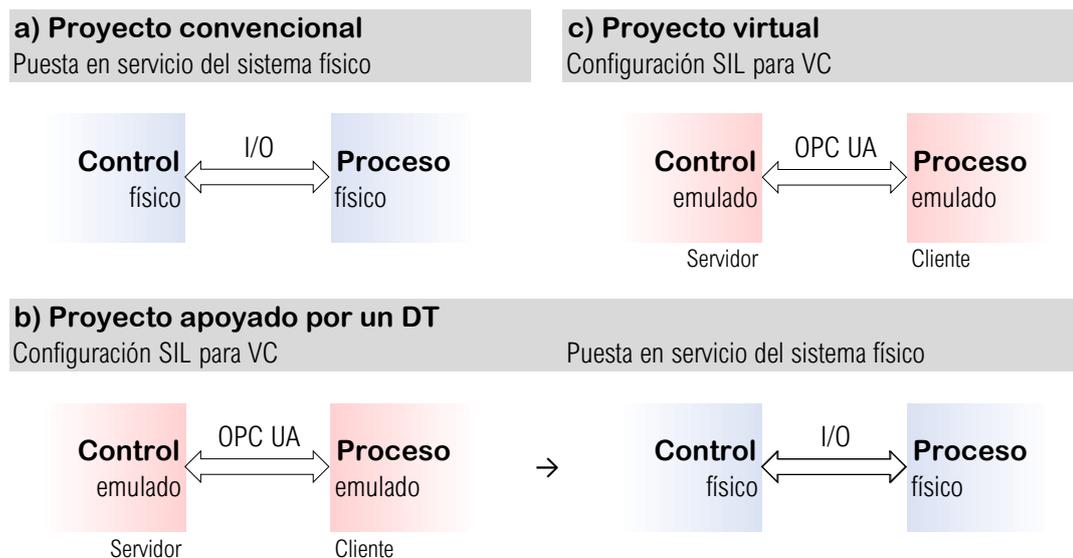


FIGURA 5.4: Configuración del sistema de control según condiciones de trabajo asignadas.

- (b) Cuatro grupos con DT y realizando la VC y posterior puesta en servicio del sistema físico. Han llevado a cabo, como los grupos de tipo *a*, el desarrollo completo de la solución automatizada, pero contando con el soporte de la emulación. Se ha seguido una configuración HIL, con un PLC físico y el proceso emulado, para su VC. Se ha tenido acceso a la célula de manufactura de manera puntual en la recta final del proyecto, exclusivamente para su puesta en marcha.
- (c) Seis grupos con DT que han trabajado en modo virtual con configuración SIL, realizado exclusivamente la VC, sin puesta en servicio del sistema físico.

5.2.2.5. Cierre del experimento

Habiendo trabajado cada uno de los grupos de alumnado según las condiciones asignadas, se ha desembocado, en el caso de aquellos a los se les solicitó hacerlo, en la puesta en marcha de la maqueta del sistema de manufactura. La actividad de cierre del experimento ha consistido en someter a las personas participantes a un cuestionario con un enfoque cualitativo con el fin de extraer los resultados para su análisis y debate. El motivo no es otro que la dificultad para cuantificar la calidad del software y el tiempo de puesta en servicio, unida a la participación de personas con diferente formación previa y no necesariamente el mismo interés por las asignaturas implicadas en la experimentación. Se ha hecho llegar al alumnado una serie de preguntas y se ha seguido una dinámica similar a la conocida como *focus group* [116]. Las preguntas han sido relativas al uso que han hecho (grupos de tipo *b* y *c*) o harían (tipo *a*) de un DT para la validación de programas de control de sistemas industriales automatizados, ya que durante el desarrollo han tenido ocasión de observar al resto de participantes.

5.2.3. Resultados

La Tabla 5.3 muestra los resultados recogidos tras los cuestionarios de cierre. Está organizada en tres partes, las relativas a cada tipo de equipo de trabajo (*a*, *b* y *c*), que

TABLA 5.3: Resultados del experimento.

<p>a) Grupos sin DT y exclusivamente puesta en servicio del sistema físico.</p> <p>Comentarios acerca de la validación del software del PLC:</p> <ul style="list-style-type: none"> - Puestas en servicio de larga duración por errores en el código. - Compleja validación a base de depuración en el entorno de desarrollo. - Dificultad para entender el sistema a controlar. - Uso elevado e ineficiente de recursos de laboratorio: maqueta. <p>Uso de un DT para la VC y en otras fases del ciclo de vida útil:</p> <ul style="list-style-type: none"> - Utilizaría la emulación en futuros proyectos. - Sería de utilidad para la detección y corrección de defectos de diseño. <p>Desarrollo de competencias técnicas mediante el uso de un DT:</p> <ul style="list-style-type: none"> - El alumnado puede tener un buen posicionamiento en el mercado laboral.
<p>b) Grupos con DT para VC y posterior puesta en servicio del sistema físico.</p> <p>Comentarios acerca de la validación del software del PLC:</p> <ul style="list-style-type: none"> - Conocimiento del sistema antes de disponer físicamente del mismo. - Posibilidad de poner a prueba el código durante su desarrollo. - Reproducción de múltiples escenarios de manera más sencilla que en el entorno. - Desarrollos más rápidos y con menos errores. Detección temprana. - La puesta en servicio es ágil y se convierte en mero trámite. <p>Uso de un DT para la VC y en otras fases del ciclo de vida útil:</p> <ul style="list-style-type: none"> - Volvería a hacer uso de la emulación en futuros proyectos. - Puede resultar de utilidad desde la fase de diseño. - Apoyo a definición de modificaciones y mejoras en las instalaciones. - Herramienta para formación del personal inexperto. <p>Desarrollo de competencias técnicas mediante el uso de un DT:</p> <ul style="list-style-type: none"> - Aprendizaje de herramientas y procedimientos emergentes en la industria. - Refuerzo de conceptos de programación y comunicaciones.
<p>c) Grupos con DT para VC exclusivamente, sin puesta en servicio.</p> <p>Comentarios acerca de la validación del software del PLC:</p> <ul style="list-style-type: none"> - Conocimiento del sistema antes de disponer físicamente del mismo. - Posibilidad de poner a prueba el código durante su desarrollo. - Reproducción de múltiples escenarios de manera más sencilla que en el entorno. <p>Uso de un DT para la VC y en otras fases del ciclo de vida útil:</p> <ul style="list-style-type: none"> - Volvería a hacer uso de la emulación en futuros proyectos. - Puede resultar de utilidad desde la fase de diseño. - Apoyo a definición de modificaciones y mejoras en las instalaciones. - Herramienta para formación del personal inexperto. <p>Desarrollo de competencias técnicas mediante el uso de un DT:</p> <ul style="list-style-type: none"> - Aprendizaje de herramientas y procedimientos emergentes en la industria. - Refuerzo de conceptos de programación y comunicaciones.

a su vez constan de tres grupos de testimonios:

- Aquellos correspondientes al uso de la emulación para la validación de código
- Las aplicaciones que se podrían desarrollar en otras fases del ciclo de vida del sistema automatizado
- Las competencias desarrolladas por el alumnado mediante el uso de estas tecnologías.

En lo que se refiere al primer bloque de resultados, las personas participantes han resaltado las virtudes de la emulación en la validación de código, tanto habiendo hecho uso de ella como habiendo observado a quienes la utilizaron. En los grupos de tipo *a*, los cuales han seguido el procedimiento convencional sin herramientas de emulación, se ha hecho hincapié en la problemática dada en los últimos años académicos y previamente descrita. Además, han considerado el DT como una herramienta útil en varios aspectos, a partir de la observación de su manejo. El alumnado con acceso a modelos virtuales ha destacado el beneficio obtenido. Han emergido ideas de gran similitud tanto habiendo realizado la puesta en servicio final (grupos de tipo *b*) como no (tipo *c*). Los grupos de tipo *b* han percibido la agilidad de dicha última fase, al contrario que en años académicos anteriores o otros grupos de tipo *a*. Cabe destacar las siguientes respuestas:

- Mejor comprensión de proceso a automatizar y del funcionamiento requerido.
- Testado de código de PLC más riguroso, reproduciendo con sencillez todos los escenarios que se puedan dar.
- Uso más eficiente del equipamiento, tras la validación virtual del software de control desarrollado.
- Menor esfuerzo en la fase de puesta en marcha, la cual se acorta en duración, al contar con un software previamente validado con el soporte de la emulación.

En lo que respecta a la aplicación de DTs en otras fases del ciclo de vida útil del sistema automatizado, el alumnado ha previsto su utilidad como soporte al diseño o en entrenamiento de personal.

Finalmente, y en cuanto al desarrollo de competencias técnicas, los profesionales del futuro participantes en el experimento han valorado el conocimiento adquirido en un ámbito que considera de utilidad y emergente en la industria.

5.2.4. Discusión

Se ha implantado con éxito el uso de tecnologías de digitalización en forma de DT en un proyecto con alumnado, en la búsqueda de una herramienta que facilite la validación de software de control en actividades académicas, y dados los casos de éxito detectados en la literatura. Se ha utilizado un DT para la verificación de código de PLC desarrollado por algunas de las personas participantes, así como para la ejecución del software del robot, el cual ha sido desarrollado previamente en otra actividad ajena al experimento.

Comparando los resultados extraídos a partir de la observación y de los testimonios del alumnado, los grupos que han utilizado la herramienta de emulación han obtenido, en comparación con aquellos que han seguido el procedimiento convencional, mejores resultados en términos de duración de la puesta en servicio, comprensión del sistema a automatizar y gestión de recursos materiales, entre otros aspectos.

A igual que en lo reflejado en la literatura, el esfuerzo previo en el modelado del DT es considerado como una inversión asumible que se traduce en desarrollo de competencias técnicas, transferencia de conocimiento y reutilización de DTs o partes de estos. Igualmente puede ser tratado el trabajo inicial del alumnado en cuanto a su familiarización con la emulación.

A partir de los resultados cualitativos del estudio se considera haber dado respuesta a la pregunta de investigación de la manera que sigue:

RQ01

¿Posibilita un DT una puesta en servicio ágil de código de PLC, con resultados de calidad y menor uso de equipamiento y riesgos asociados?

El uso de un DT aporta mejoras en términos de comprensión del trabajo a realizar y conocimiento del sistema a automatizar; agilidad en el desarrollo, la validación y la puesta en servicio del sistema de control; rigor y creatividad; y disponibilidad de equipamiento.

Se ha buscado una solución a la problemática a la que se enfrentan los colectivos de alumnado y de profesorado en la impartición de contenidos sobre automatización industrial en el aula académica. Se puede afirmar haberla encontrado, y se basa en la integración de la emulación en los planes de estudios. Cabe destacar la ausencia de problemas en la experimentación aunque sí alguna debilidad en el método empleado:

- La dificultad en poder cuantificar con precisión las mejoras relativas al uso de un DT. Se ha adoptado un enfoque cualitativo que ha traído consigo unos resultados que se consideran concluyentes y representativos.
- El hecho de que no haya emergido una de las principales desventajas del uso de la emulación en este ámbito, ampliamente mencionada en la literatura: la complejidad del modelado. En esta actividad no se contaba con el modelado como tarea a acometer ni objetivo.

5.3. Trabajo relacionado

Concluido este primer estudio, se ha procedido a la divulgación de sus resultados en un artículo de conferencia [117] y un capítulo de libro [118] (ver sección 8.2.1). Además, este experimento ha traído consigo trabajo relacionado tanto internamente como externamente a MU-GEP, posicionando al grupo de investigación en el ámbito de las tecnologías utilizadas:

- En primera instancia, ha derivado en la incorporación de estas tecnologías en asignaturas de automatización industrial en niveles de Ciclo Formativo de Grado Superior (CFGS), Grado y Máster. Se han desarrollado casos no sólo en en S3D, sino que además se han utilizado las herramientas CIROS Education 6.2, Factory I/O v2.2.3 y Simumatik Platform.
- Además, ha suscitado el interés de centros de formación. Sirva como muestra la colaboración con Tknika (ver sección 8.2.2).
- Todo ello ha dado pie a los trabajos que se muestran en los Capítulos 6 y 7, y en el Anexo A.

5.4. Conclusión y trabajo futuro

Se ha propuesto la digitalización para mejorar la formación en automatización industrial en los términos descritos, en respuesta a problemas que se venían dando en el aula y/o laboratorio. Como primer paso, se ha diseñado un experimento consistente en el desarrollo de una solución automatizada para un sistema de manufactura, en el que a parte del alumnado se le ha facilitado un DT del proceso, para testar tanto la validez de la herramienta como los resultados detectados en la revisión de la literatura.

Los resultados cualitativos demuestran la validez de la tecnología propuesta para evitar situaciones que se venían dando en el ámbito mencionado, y que se han vuelto a reproducir en aquella parte del alumnado que no ha tenido a mano un modelo virtual del sistema. Dichos resultados, así como la constitución del marco de trabajo y la adquisición de competencias, han abierto la puerta a próximos casos de uso en otras tareas, ámbitos o fases del ciclo de vida de un sistema industrial automatizado.

Capítulo 6

Implementación semi-compilada de redes de Petri en PLCs.

Los sistemas de tipo **DEDS** suelen ser modelados mediante **PNs**, las cuales tienen la capacidad de representar aspectos como la concurrencia, la sincronización y los recursos compartidos, al contrario que un **GRAFCET**, caso particular de una **PN**. Sin embargo, no existe una forma sistemática eficaz de implementar una **PN** en un **PLC**, por lo que es muy común hacerlo en algún software externo que se comunique con el dispositivo de control. Hasta la fecha, los intentos de implementación han dependido de cómo se programe la lógica de lugares y transiciones para cada aplicación.

En este capítulo se presenta una novedosa metodología de implementación de **PNs** independiente de la aplicación y de la plataforma. Se realiza de forma sistemática dentro de **PLCs** industriales. Una gran parte del código se valida automáticamente antes del desarrollo del sistema de control, ya que la estructura de la red y la evolución del marcado son testados a partir del análisis estructural del modelo **PN**. Únicamente la interpretación de la red es programada manualmente, limitándose exclusivamente a esta parte las posibles fuentes de error. Por tanto, esta metodología representa un método de implementación de **PNs** sistemático y semi-compilado.

Se muestra un caso de uso apoyado en un **DT** de un sistema de fabricación específico y no disponible físicamente, cuyas características lo hacen ejemplo típico de aplicación de **PNs**. La solución automatizada requerida por dicho proceso es desarrollada a partir de varios enfoques de implementación de **DEDSs**, incluido el objeto de estudio, y ejecutada en dos dispositivos diferentes. Se somete a prueba la portabilidad del marco propuesto y se comparan los periodos de ciclo de scan requeridos por cada aplicación.

6.1. Introducción

El trabajo que se describe a continuación ha sido llevado a cabo en el contexto de la actividad de I+T, concretamente en la búsqueda de una metodología para programar **DEDSs** de una manera ágil, sencilla y con mínima probabilidad de errores en el código del **PLC**.

6.1.1. Sistemas dinámicos de eventos discretos

Un **DEDS** es un sistema dinámico y asíncrono en el que las transiciones entre los diferentes estados del sistema se producen como consecuencia de la ocurrencia de

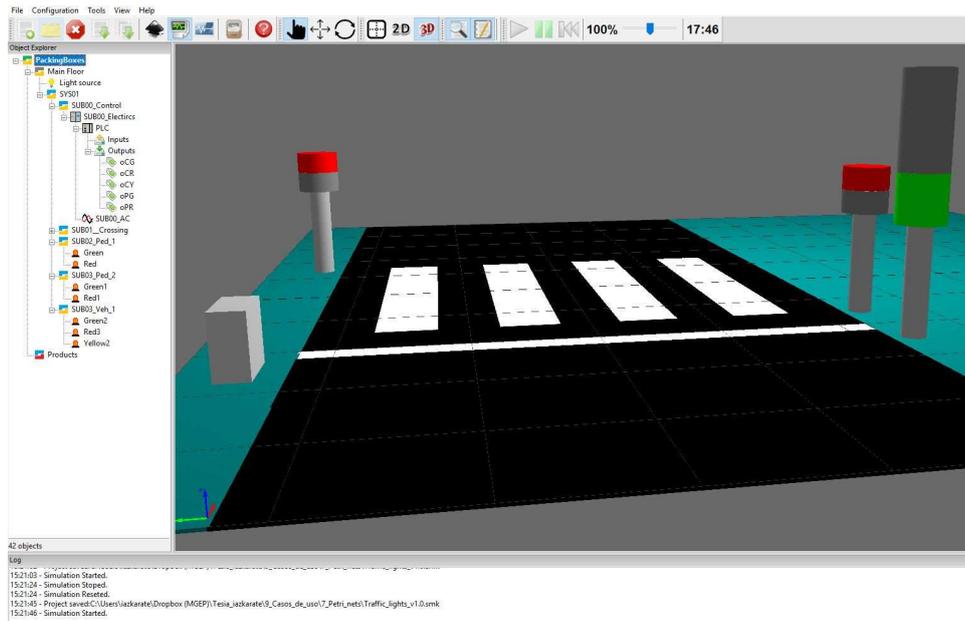


FIGURA 6.1: Gemelo digital de un sistema dinámico de eventos discretos: paso de peatones regulado por semáforos.

ciertos eventos en instantes discretos de tiempo [5]. Su modelado se realiza mediante autómatas de estados finitos con eventos parcialmente observables junto con un mecanismo para la activación y desactivación de un subconjunto de transiciones de estado [119]. Este tipo de sistemas está presente tanto en la vida cotidiana como en la industria manufacturera:

- La sincronización de semáforos en un paso de peatones, tras la expiración de cada temporización (ver Figura 6.1).
- Personas que se incorporan a una cola de espera para acceder a un mostrador, y la abandonan tras haber realizado la gestión oportuna.
- La atención, por parte de un camarero, a dos clientes de un establecimiento de hostelería, anotándose/sirviéndose los pedidos en cualquier orden.
- Sistemas de fabricación flexibles y/o discretos, controlados por robots y máquinas programables y con características como la evolución concurrente, sincronización y/o recursos compartidos (como el ya mostrado en la Figura 1.3).

Se acostumbra a modelar **DEDSs** a partir de una de estas aproximaciones: **PN** o **GRAFCET**. Una **PN** es considerada como un método eficaz para el modelado y control de dichos sistemas, ya que permite representar sistemas que se caracterizan por ser concurrentes, asíncronos, distribuidos, paralelos, no deterministas y/o estocásticos [7]. En el ámbito de la automatización industrial, el marco más generalizado es el **GRAFCET**, herramienta gráfica muy utilizada para la programación de **PLCs** [3] y considerada como un caso particular de una **PN** [87]: funcionalidades que están integradas en la propia estructura de una **PN** son más laboriosas de implementar en un **GRAFCET**. Las siguientes secciones detallan las particularidades de ambos enfoques y cómo se implementan en controladores.

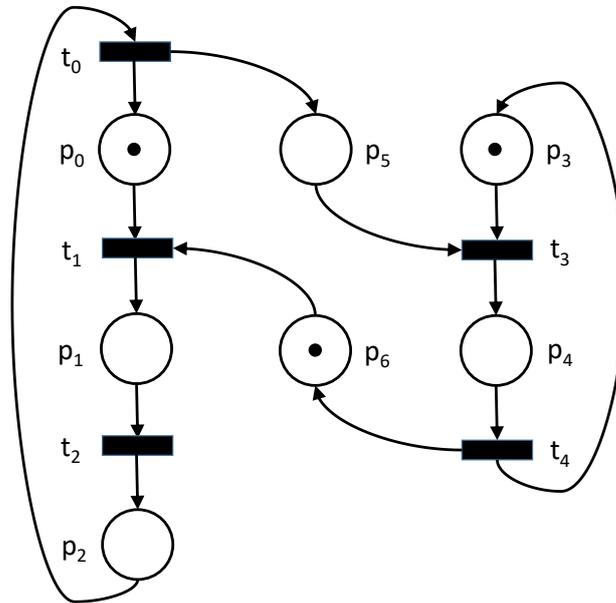


FIGURA 6.2: Estructura de la red de Petri para el paso de peatones regulado por semáforos.

TABLA 6.1: Interpretación a incorporar a la estructura de la red de Petri para el paso de peatones regulado por semáforos.

Lugar	Acción	Descripción
p_0	oCR	Luz roja para vehículos.
p_1	oCG	Luz verde para vehículos, y temporización.
p_2	oCY	Luz ámbar para vehículos, y temporización.
p_3	oPR	Luz roja para peatones.
p_4	oPG	Luz verde para peatones, y temporización.
p_5	-	Marca de activación de la luz verde para peatones.
p_6	-	Marca de activación de la luz verde para vehículos.
Transición	Condición	Descripción
t_0	tVY	Fin de temporización de luz ámbar para vehículos.
t_1	-	Directa.
t_2	tVG	Fin de temporización de luz verde para vehículos.
t_3	-	Directa.
t_4	tPG	Fin de temporización de luz verde para peatones.

6.1.2. Modelado de sistemas dinámicos de eventos discretos

6.1.2.1. Modelado basado en redes de Petri

Las PNs son consideradas como un formalismo adecuado para el modelado, el control y la visualización del comportamiento de DEDSs [10]. Hacen posible la representación de sistemas concurrentes, asíncronos, distribuidos, paralelos, no deterministas y/o estocásticos [7]. Se trata de una herramienta gráfica visual que combina un diagrama de bloques y marcas que simulan la dinámica y las actividades concurrentes de los sistemas. Desde su proposición por parte de [120] se dispone de un amplio cuerpo de teorías para el análisis algebraico de su comportamiento. Se han aplicado las PNs con éxito en sectores como el ferroviario [121], en satélites [122], o en logística y sistemas de transporte [123]. Además, [8] las define como un método eficaz para el diseño, la evaluación de prestaciones y el control de sistemas de fabricación flexible [9].

6.1.2.1.1 Definición del grafo

Una PN está compuesta [6] por una estructura (sirva como ejemplo la Figura 6.2 para el cruce de semáforos) y una interpretación asociada a la misma (Tabla 6.1), así como su dinámica. En términos estructurales, [10] describe un grafo PN como un gráfico bipartito que contiene dos tipos de nodos: lugares (identificados por una $p \in P$ y representados mediante círculos) y transiciones ($t \in T$, mediante barras). Ambos tipos de nodos son interconectados mediante arcos dirigidos y dotados de peso. Una red Lugar/Transición o Place/Transition (P/T) es designada por [6] como $N = \langle P, T, Pre, Post \rangle$, donde P y T son los conjuntos de lugares y transiciones, y Pre y $Post$ son las matrices de incidencia de dimensiones $|P| \times |T|$ y con valores naturales. Por ejemplo, $Post[p, t] = w$ significa que hay un arco de t a p con peso o multiplicidad w . La ausencia de arco implica un valor nulo en la matriz. Para el ejemplo objeto de estudio, dichas matrices son como sigue. Nótese que todos los pesos son unitarios, lo que caracteriza a una red ordinaria:

$$Pre = \begin{matrix} & \begin{matrix} t_0 & t_1 & t_2 & t_3 & t_4 \end{matrix} \\ \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix} & \begin{matrix} p_0 \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \end{matrix} \end{matrix}$$

$$Post = \begin{matrix} & \begin{matrix} t_0 & t_1 & t_2 & t_3 & t_4 \end{matrix} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} & \begin{matrix} p_0 \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \end{matrix} \end{matrix}$$

El marcado es un vector de dimensión $|P|$ y con valores naturales. Una red marcada, un sistema P/T o una PN es un par $\langle N, m_0 \rangle$ donde m_0 es el marcado inicial. Se representa transpuesto:

$$m_0^t = \begin{pmatrix} p_0 & p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

6.1.2.1.2 Dinámica del sistema

La dinámica de una PN se caracteriza por la distribución de marcas (o *tokens*, o *jetons*) a lo largo de los lugares, y la evolución de la misma. Dicho movimiento depende del disparo controlado de transiciones, es decir, de su sensibilización y del cumplimiento de las condiciones de disparo asociadas.

6.1.2.1.3 Condiciones para el disparo de una transición

Para que una transición sea disparable deberá darse (i) la sensibilización de la misma y (ii) el cumplimiento de la condición (booleana) asociada en la interpretación. Una transición t es sensibilizada por el marcado m si $m \geq Pre[P,t]$. No toda transición disparable ha de hacerlo, y en un instante determinado no puede dispararse más de una de manera simultánea.

6.1.2.1.4 Determinación del nuevo marcado

El disparo de una transición produce un nuevo marcado $m' = m + C[P,t]$ donde $C = Post - Pre$ es la matriz de incidencia o de flujo de marcas de la red.

$$C = Post - Pre = \begin{pmatrix} t_0 & t_1 & t_2 & t_3 & t_4 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 1 & -1 \\ 1 & 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 & 1 \end{pmatrix} \begin{matrix} p_0 \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \end{matrix}$$

6.1.2.1.5 Validación estructural

Como se ha aludido con anterioridad, existe un amplio cuerpo de conocimiento en torno a la PN. Distintos autores proponen extensiones a la dinámica en el disparo de transiciones: redes continuas e híbridas [124]; redes de alto nivel como una *red Predicado-Transición* o *Predicate/Transition net (PrT-net)* [125] o una *red de Petri coloreada* o *Colored Petri net (CPN)* [126]; y aproximaciones para diferentes tipos de interpretación (estocástica [127]).

En otros trabajos se proponen técnicas para la validación de los modelos y/o análisis de sus prestaciones. Para ello se ha definido una serie de propiedades básicas a cumplir [7] [128] [129] [130]:

- La vivacidad, relacionada con la posibilidad de alcance de cualquier estado del sistema, y en particular de estados de bloqueo que no permitan su posterior evolución.
- La ciclicidad, relacionada con la posibilidad de retorno desde cualquier estado del sistema al inicial.
- La limitación, relacionada con la finitud en el número de estados del sistema.

Hay disponibles en el mercado herramientas software con funciones como la edición gráfica de la red, la simulación de la evolución del marcado, el análisis de propiedades, y la generación de la definición formal de la red a partir de su representación matricial.

6.1.2.1.6 Interpretación de la red

La interpretación de la red consta, por un lado, de eventos modelados en transiciones, los cuales provocan un cambio en el marcado. Por otro, se asocian acciones a lugares. Esto puede consistir en (i) tiempo (constante, aleatorio) para evaluación del rendimiento; (ii) condición acción de control. La interpretación correspondiente al ejemplo es la ya referida y mostrada en la Tabla 6.1.

6.1.2.2. Implementación de redes de Petri en sistemas industriales automatizados

El PLC, dispositivo de control para DEDSs basado en información de proceso y operador [3], es ampliamente utilizado en entorno industrial. Esto se debe a las características que lo diferencian de un ordenador convencional [4]: (i) ejecución cíclica del programa de control, (ii) fiabilidad, (iii) adaptación a entornos con presencia de ruido eléctrico, vibraciones, temperatura y/o humedad extrema, y (iv) mantenimiento sencillo.

No se cuenta con una manera sistemática y de uso generalizado para la implementación de PNs en PLCs. Es común realizarla en un controlador externo a este, mediante un software que se comunica con el PLC. Los desarrollos en PLC dependen de cómo se programa la lógica de lugares y transiciones. La ausencia de un método hace que pueda haber tantas formas de implementación como programadores, y cualquier modificación en la PN requiere una regeneración del código. Aunque ha habido intentos de sistematización a partir de métodos para programar la lógica de lugares y transiciones, no han sido ampliamente adoptados.

La literatura presenta trabajos acerca de metodologías de implementación de PNs en PLCs de forma interpretada, es decir, con la gestión tanto de la estructura como del marcado integrada en el propio código. El método de [88] para la generación de LDs, los cuales son ampliamente utilizados en el control de eventos discretos, agiliza el desarrollo, la depuración y la reingeniería. Está basado en modelos PN válidos para el análisis formal de sistemas, así como para verificar la vivacidad y la ciclicidad. Evidencia, asimismo, la necesidad de construir PNs de manera simple, poder evaluar sus propiedades y simular la evolución del marcado. En un entorno en el que los sistemas son cada vez más complejos, la revisión de [89] acerca de los enfoques para la conversión de PNs como solución integrada a LDs hace hincapié en la limitación de este tipo de implementaciones, y en la necesidad de herramientas para el control, análisis, evaluación y simulación de sistemas. El conjunto de reglas

de [10] para la aplicación de **LDs** supera los desarrollos *ad hoc* meramente intuitivos. Otra revisión, la de [90], hace referencia a este tipo de conversiones, entre otras implementaciones de controladores lógicos. Hay, además, reseñas al uso de otros lenguajes de programación, más allá de los mencionados **LDs** [91] [92] [93] [94]. Se trata de listados de instrucciones [95] [96] o propuestas más recientes en lenguajes de programación estándar [97] o **ST** [98], para la portabilidad entre **PLCs**. En un artículo reciente, [99] presenta la conversión automática a código de **PLC** a partir de la edición de la estructura y la interpretación de la red en un entorno gráfico propio.

Pero las metodologías identificadas presentan algunas debilidades. Se trata de desarrollos interpretados, totalmente dependientes de la aplicación y de quien la programe. La carga de trabajo en el desarrollo de código es notable y este es fuente de posibles errores. Además, cualquier modificación en la estructura o en la interpretación de la red conlleva la reescritura o la regeneración del código. En cuanto a la validación de la **PN**, se realiza al final del desarrollo, con el equipamiento ya montado, o a través de un **DT** que lo emula, o mediante el convencional forzado de valores en las entradas y la observación del comportamiento de las salidas en el propio entorno de desarrollo del **PLC**.

6.1.2.3. Modelado basado en metodología GRAFCET

GRAFCET es una herramienta gráfica de uso habitual en la programación del **PLCs** [3]. Se trata de un caso particular de una **PN**. Consiste en una red salva [87], es decir, se garantiza la propiedad de limitación al ser el 0 ó 1 marcado de cualquier lugar de la **PN**, $\forall p \in P, m(p) \leq 1$.

6.1.2.4. Implementación de GRAFCET en sistemas industriales automatizados

Tradicionalmente se ha asociado una variable booleana a cada etapa [11] [12], evaluándose su estado en cada ciclo de scan del **PLC**. En la actualidad, numerosos entornos de desarrollo incluyen **SFC**, un método de modelado y descripción gráfica para sistemas secuenciales, adecuado para **GRAFCET** y estandarizado en la norma IEC 61131-3.

6.1.3. Comparativa entre ambos enfoques

Se puede afirmar que un enfoque **PN** es más adecuado que un **GRAFCET** a la hora de controlar un **DEDS**. Por una parte, la definición matemática formal (C, m_0) de una estructura de **PN**, y los consiguientes métodos de validación formal, hace posible testar condiciones necesarias en la estructura de la misma. Los métodos de validación son más potentes que aquellos que se proponen para **GRAFCET**, limitados al uso correcto de las convergencias para el cierre de divergencias previamente abiertas [87].

Por otro lado, ciertas particularidades que puedan darse en un sistema están integradas en la estructura de una **PN** y pueden ser validadas de forma directa, y en cambio pueden obligar al uso de variables internas y código en un **GRAFCET**. Esto puede dar pie a desarrollos de mayor duración y con más probabilidad de contener errores. Las citadas particularidades son las siguientes:

- En un **GRAFCET** pueden modelarse recursos compartidos, siempre que sean unitarios. En caso contrario, se hace uso de la interpretación. En una **PN** forman parte de la estructura, en cualquier caso.

- Al igual que en una **PN**, en un **GRAFCET** pueden modelarse sincronizaciones sin tener que recurrir a la interpretación. Se hace a partir de divergencias/convergencias en Y .
- Al contrario que en una **PN**, lo que no puede realizarse estructuralmente en un **GRAFCET** es modelar capacidades (finitas, limitadas) de *buffers*. Necesariamente se recurre a la interpretación.

En términos generales, todo aquello que precise de una cifra no unitaria para su representación, es decir, más allá del dato booleano, no puede integrarse en la estructura de un diagrama **GRAFCET**, por lo que hay que gestionarlo mediante la interpretación.

6.1.4. Definición del problema

El personal técnico a cargo de implementaciones de **DEDSs** en **PLCs** recurre al **GRAFCET** como herramienta de modelado por lo visual de la programación en **SFC** y por la no existencia de una metodología para dichos controladores. A pesar de sus virtudes, la **PN** no es una herramienta de uso generalizado. Se entiende que un cambio de paradigma en este tipo de implementaciones puede contribuir a sistematizar y simplificar desarrollos y, en consecuencia, acortar plazos de entrega de soluciones automatizadas.

Se propone una metodología para implementar una **PN** en un **PLC** de forma sencilla, haciendo uso de la mencionada definición matemática formal de una red, que permite validar automáticamente parte del programa del dispositivo de control, minimizando la carga de trabajo y, en consecuencia, los puntos susceptibles de error. Así, una representación matricial de **PN** previamente validada se copia directamente en la memoria de datos de un **PLC**. Un bloque de programa fijo gestiona el marcado de la red, esto es, la sensibilización de las transiciones, calculando el nuevo marcado alcanzado tras sus posibles disparos. Realizado y validado el modelo, se procede a codificar el programa. Dicha tarea se limita a la interpretación de la red, es decir, a la asignación de condiciones (consultas de entradas) a transiciones y de acciones (escritura de salidas) a lugares. De este modo, los proyectos pueden llevarse a cabo y ponerse en marcha de forma muy eficiente. Esto supone una implementación de **PNs** semi-compilada en lugar de una exclusivamente interpretada. Nótese que en el contexto de este trabajo se utilizan **PNs** simples. La interpretación se circunscribe a condiciones (eventos) y acciones asociadas a transiciones y lugares, respectivamente. Se realiza un análisis estructural básico sin evaluación previa del rendimiento mediante otra interpretación temporal y/o estocástica.

La Figura 6.3 ilustra la implementación propuesta, la cual puede contribuir al estado del arte en varios aspectos:

- El modelo de **PN** es construido, evaluado y simulado con sencillez, como demandan autores como [88]. Se utiliza una herramienta software visual e intuitiva para la generación de datos estructurales que son gestionados por bloques de programa fijos, y para la validación de parte de la implementación. Mientras que la herramienta presentada en [99] es exclusivamente para la edición de la estructura e interpretación de la **PN**, en este trabajo se propone el uso de un software comercial que proporciona la edición de la estructura, la simulación y la validación.

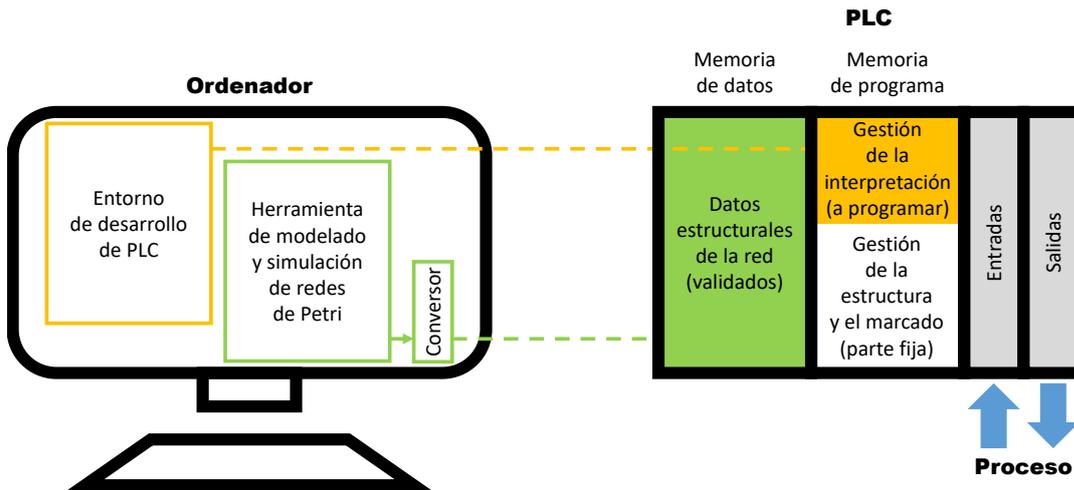


FIGURA 6.3: Implementación basada en redes de Petri: ordenador y PLC.

- La programación y validación manual se limita a la parte de interpretación de la red.
- Cualquier cambio en la estructura de la red o en el marcado inicial es directo e impacta sólo en el área de datos del controlador, nunca en el programa, que permanece sin modificar. En cambio, la literatura muestra metodologías de generación de código, transformando **PNs** completas (estructura e interpretación) a lenguajes de programación estándar de **PLC**, como [107]. Este automatiza y acelera la conversión a **LDs** a partir de matrices características, pero sin generarlas automáticamente ni utilizar reglas de validación, por lo que cualquier cambio implica que todo el código sea regenerado para su validación manual.

El personal técnico que afronta este tipo de proyectos en el contexto de su actividad investigadora se encuentra con la necesidad de testar nuevas implementaciones contra sistemas industriales automatizados no disponibles, por ser muy específicos para la realización de pruebas y reproducción de distintos escenarios. No contar con el sistema físico hace que el código desarrollado se verifique de manera convencional, es decir, a partir del forzado de entradas del dispositivo de control en su propio entorno de desarrollo. La validación de sistemas de cierta complejidad se convierte en una fase larga, dificultosa y no está garantizada. En este caso, se ha requerido someter a prueba la metodología objeto de estudio contra un sistema de fabricación flexible reconocido en la literatura como ejemplo típico en el que afloran las virtudes de una **PN**. Disponer del mismo emulado a través de su **DT** ha apoyado continuamente el desarrollo y su posterior testado y validación de una manera visual y ágil.

6.2. Evaluación empírica

Para la evaluación de este trabajo se ha procedido como se detalla a continuación. Tras enunciar las **RQs**, se ha caracterizado el ensayo realizado y se han obtenido y discutido los resultados.

6.2.1. Preguntas de investigación

A partir de la problemática descrita, de lo extraído de la revisión de la literatura, y la propuesta realizada en este capítulo, se han enunciado las siguientes RQs:

RQ02

¿Es el enfoque semi-compilado y basado en PNs más sencillo de programar que los ya existentes para DEDSs en PLCs?

RQ03

¿Requiere el enfoque objeto de estudio más recursos del PLC en términos de período de ciclo de scan?

RQ04

¿Es el enfoque objeto de estudio independiente de plataforma y sencillo de portar, tanto código como datos?

RQ05

¿Es el DT una herramienta que facilita la validación de desarrollos genéricos contra sistemas industriales no disponibles físicamente?

6.2.2. Caracterización del ensayo

Esta sección está compuesta de cinco bloques. En el primero de ellos se enumera el material requerido. Posteriormente se detallan las características del sistema emulado que se ha utilizado en el experimento. Se sigue con el trabajo realizado y las condiciones en las que se ha llevado a cabo, así como la metodología seguida. Se cierra el ensayo con la obtención de resultados.

6.2.2.1. Material utilizado

El material necesario para un trabajo de estas características es el que sigue. Se especifican herramientas y versiones concretas de las que se ha hecho uso:

- Dispositivos de control de sistemas industriales automatizados.
Se ha recurrido a PLCs de dos fabricantes: (i) SIMATIC S7-1500 CPU 1512C-1 PN de Siemens, y (ii) CPU NX102-1020 de Omron.
- Herramientas software:
 - Plataformas para desarrollo y emulación de software de control: (i) TIA Portal V15.1 de Siemens para programación y PLCSIM Advanced V2.0 SP1 para emulación, y (ii) Sysmac Studio 1.25 de Omron para programación y emulación.
 - Una herramienta de emulación de sistemas y procesos industriales.

Un DT en S3D ha apoyado el desarrollo, la simulación y la validación del software de control.

- Una herramienta de modelado y análisis de PNs.

Se ha utilizado *Platform Independent Petri net Editor 2 (PIPE2)* [131] [132] para la edición estructural de la red, verificación de condiciones necesarias y generación de matrices representativas (Figura 6.4).

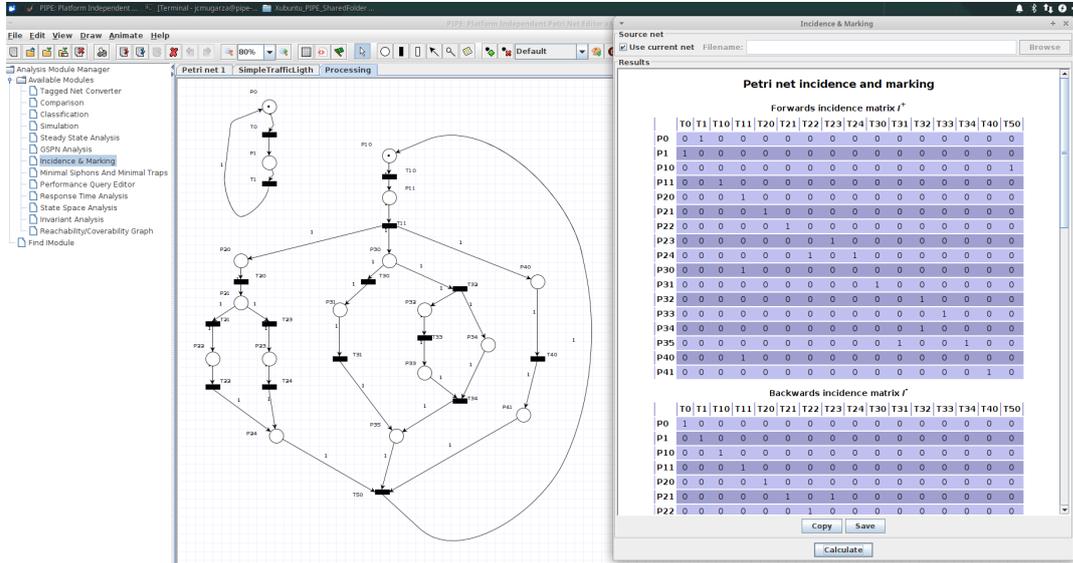


FIGURA 6.4: Herramienta de modelado y análisis de redes de Petri.

- Además, se ha desarrollado expresamente una aplicación para la conversión de los datos generados por PIPE2, en formato HTML, a formato texto, de cara a su adición a los módulos de declaración de variables a varios entornos de desarrollo, entre ellos los ya mencionados (Figura 6.5).

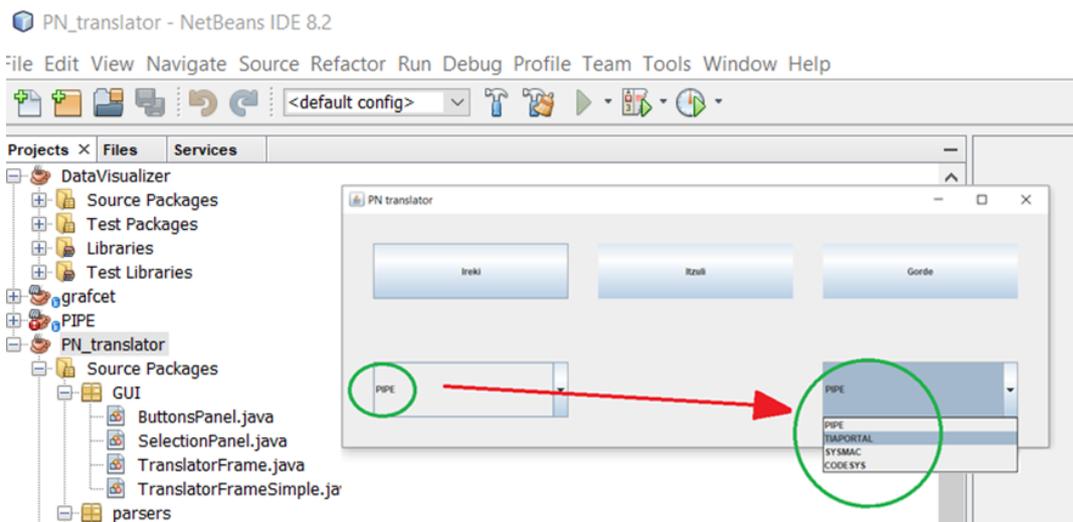


FIGURA 6.5: Software para la conversión de formato de los datos.

6.2.2.2. Caracterización del modelo virtual

En esta sección se introduce al lector a las características con las que cuenta el DT con el que se ha llevado a cabo este experimento.

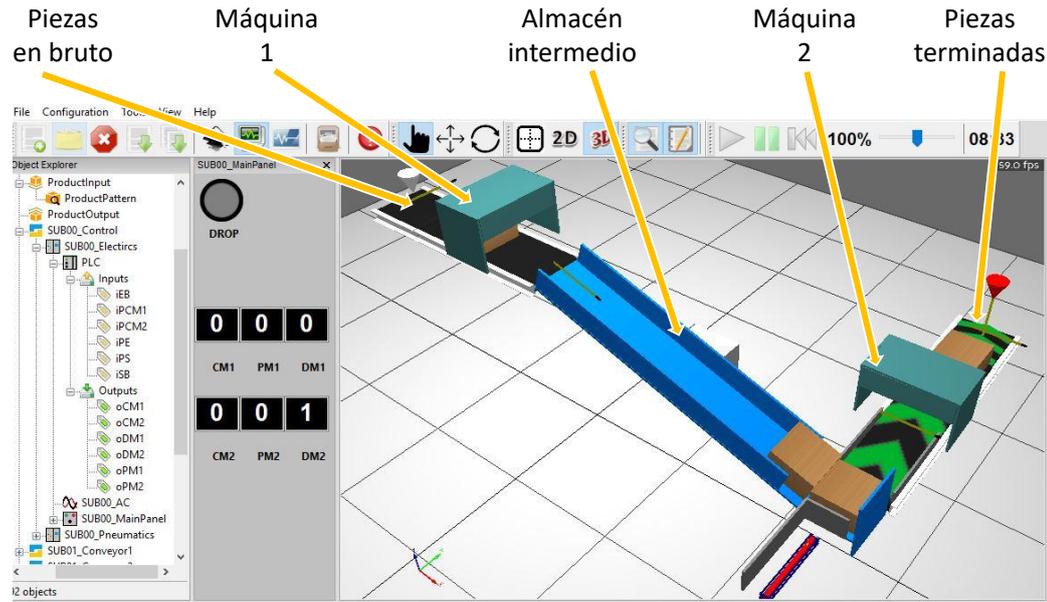


FIGURA 6.6: Gemelo digital de la célula de manufactura.

6.2.2.2.1 Sistema objeto de estudio

Se ha emulado la célula de fabricación flexible [6] previamente descrita y mostrada en la Figura 1.3. Se compone de dos máquinas por las que las piezas pasan siempre en idéntico orden, con un almacén intermedio de capacidad limitada entre ellas. Un robot carga y descarga dichas estaciones, realizando cada vez únicamente una de las cuatro operaciones posibles al ser recurso compartido.

6.2.2.2.2 Diseño del modelo virtual

Se ha utilizado [S3D](#), de licencia gratuita y en cuyo manejo se cuenta con la experiencia necesaria. El diseño lo ha realizado el doctorando durante el experimento descrito en este capítulo, y previamente a las pruebas requeridas.

6.2.2.2.3 Particularidades del modelo virtual

La Figura 6.6 muestra el modelo virtual utilizado, el cual cuenta con el cableado virtual del sistema, incluyendo el de sensores/(pre)actuadores con entradas/salidas del objeto PLC.

El DT tiene una serie de particularidades. Se ha optado por no emular el robot para no tener que programarlo, y las señales de control al mismo se convierten en movimientos de un sistema de cintas transportadoras que realizan la carga y descarga de las dos estaciones. El almacén intermedio consiste en una rampa en cuya parte inferior hay un pistón que adelantado contiene las piezas y retrasado deja caer una por gravedad. El mecanizado consiste en mantener una señal activada durante un tiempo, estando la máquina correspondiente cargada con una pieza. El emulador incluye un visor para monitorizar esas seis señales y verificar que no se da más de una señal de carga o descarga a la vez, al ser el robot un recurso compartido.

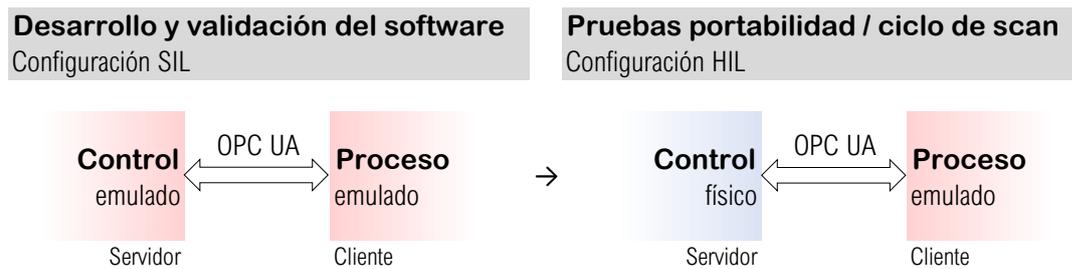


FIGURA 6.7: Configuración del sistema de control.

6.2.2.2.4 Conectividad

Para la conexión de señales del dispositivo de control con el DT a través de OPC UA se han tenido en cuenta varios aspectos. En el caso del PLC, los siguientes:

- Se ha habilitado el servidor OPC UA de la CPU en una dirección IP y un puerto.
- Se ha asignado a todas las variables compartidas el mismo nombre que en el emulador.
- Se ha desactivado la actualización automática de memorias asociadas a entradas, para su escritura por parte del modelo virtual.

En lo que respecta al DT:

- En el objeto PLC del DT se hace referencia a la citada dirección IP y puerto del servidor OPC UA.

6.2.2.2.5 Configuración

Se han dispuesto sistema de control y proceso emulado según una configuración SIL para desarrollo y validación de la propuesta, y HIL para pruebas de portabilidad y mediciones de período de ciclo de scan (Figura 6.7).

6.2.2.3. Trabajo realizado

Se han llevado a cabo varias implementaciones de DEDSs en PLCs de dos fabricantes distintos, a partir del enfoque objeto de estudio y de otros similares y de carácter interpretado. Este planteamiento es debido al interés por (i) testar la portabilidad de la propuesta y (ii) compararla con el resto de casos en términos de recursos consumidos. Los enfoques seguidos en estas pruebas son estos:

- La propuesta semi-compilada, y desarrollada en ST.
- Una implementación que procesa estructura e interpretación de una PN, a partir de estructuras IF THEN en ST.
- Un desarrollo que procesa estructura e interpretación de una PN, a partir de la activación y desactivación (set-reset) de variables booleanas en LDs.
- Una implementación basada en SFC, método gráfico de modelado y descripción, adecuado para GRAFCET y disponible en el entorno TIA Portal de Siemens. La Figura 6.8 muestra el diagrama correspondiente al caso objeto de estudio.

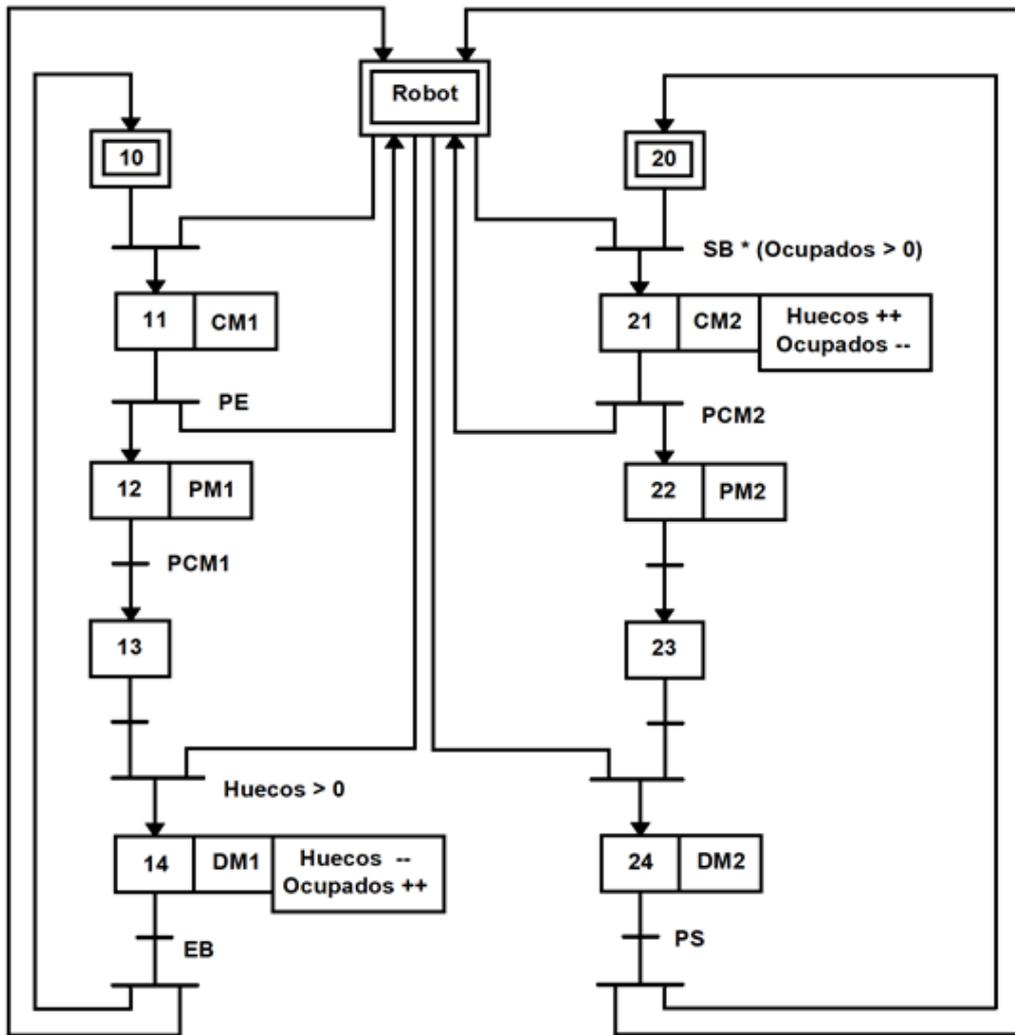


FIGURA 6.8: GRAFCET para la célula de manufactura.

El trabajo realizado ha conestado de las siguientes fases:

1. Definición:

- Especificaciones del algoritmo.
- Especificaciones de la metodología de implementación de [DEDSs](#) en [PLCs](#).

2. Diseño:

- Diseño del [DT](#) del sistema de fabricación objeto de estudio.
- Modelado y análisis estructural, en la herramienta [PIPE2](#), de la [PN](#) correspondiente al funcionamiento deseado.

3. Primera implementación del enfoque, para un [PLC](#) Siemens:

- Preparación de una herramienta propia para la adaptación de la representación matricial de la estructura de la [PN](#) generada en [PIPE2](#) a la sintaxis de la zona de variables de diferentes entornos de desarrollo.

Algorithm 1: Disparo de transiciones y actualización de marcado en una red de Petri general.

$i = 1, \dots, m$; donde m es la cardinalidad de P ; $\wedge j = 1, \dots, n$; donde n es la de T .

inicialización del vector de marcado: $x(p_i) := x_0(p_i)$;

inicialización de la matriz de incidencia: $Inc(p_i, t_j) := Post(p_i, t_j) - Pre(p_i, t_j)$;

while *TRUE* **do**

 crear lista de transiciones sensibilizadas $E(t_j)$:

 una transición está sensibilizada $\iff x(p_i) \geq Pre(p_i, t_j), \forall p_i \in I(t_j)$;

$I(t_j)$ es el conjunto de lugares de entrada de una transición $t_j \in T$.

 computar lista de condiciones: $Cond(t_j), \forall t_j \in E(t_j)$;

 crear lista de transiciones disparables $F(t_j)$:

 una transición es disparable $\iff Cond(t_j) = TRUE, \forall t_j \in E(t_j)$;

if $\exists t_j / F(t_j) \neq 0$ **then**

 decidir qué transición $t_j \in F(t_j)$, si hay, debería ser disparada;

if se decide disparar $t_j \in F(t_j)$ **then**

 calcular nuevo vector de marcado: $x'(p_i) := x(p_i) + Inc(p_i, t_j)$;

 actualizar vector de marcado: $x(p_i) := x'(p_i)$;

 actualizar acciones: $\forall p_i / x(p_i) > 0 \implies Acc(p_i) := TRUE$;

end

end

end

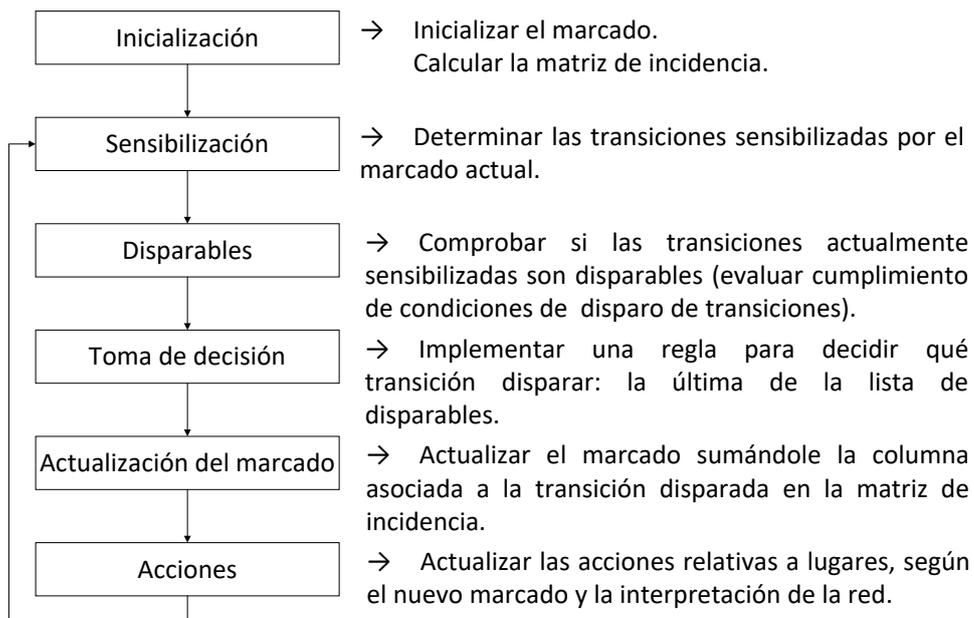


FIGURA 6.9: Flujo de programa del algoritmo.

- Creación de un nuevo proyecto en entorno TIA Portal, asignación del hardware a utilizar y declaración de las variables requeridas por el algoritmo para su correcto funcionamiento.
 - Desarrollo del código para el tratamiento de la estructura de la red y la evolución de su marcado, así como de la interpretación de la misma.
 - Transferencia de la caracterización de la estructura de la **PN** a memoria de datos del **PLC**, mediante el software propietario.
 - **VC** de la aplicación y mediciones del período del ciclo de scan del dispositivo de control.
4. Segunda implementación del enfoque, para un **PLC** Omron:
- Creación de un nuevo proyecto en entorno Sysmac Studio, asignación del hardware a utilizar y declaración de las variables requeridas por el algoritmo para su correcto funcionamiento.
 - Portabilidad del código de la primera implementación.
 - Transferencia de la caracterización de la estructura de la **PN** a memoria de datos del **PLC**, mediante el software propietario.
 - **VC** de la aplicación y mediciones del período del ciclo de scan del dispositivo de control.
5. Implementación de los otros enfoques de carácter totalmente interpretado, para ambos **PLCs**.

6.2.2.4. Metodología

Para la implementación de **PNs** en **PLCs** que se propone en este experimento, a continuación se caracterizan tanto el algoritmo en el que se apoya como el procedimiento a seguir para el desarrollo de la solución automatizada que responda a las especificaciones de funcionamiento.

6.2.2.4.1 Caracterización del algoritmo

El algoritmo se caracteriza por (i) qué hace (ver Algoritmo 1), (ii) qué flujo de programa tiene (ver Figura 6.9), y (iii) cómo se utiliza. En esta sección se describen los dos primeros aspectos, quedando el tercero para la siguiente. El núcleo del algoritmo implementa el disparo de transiciones y las reglas de actualización del marcado de un **PN** general. Disponiendo, en un bloque de datos del **PLC**, de las matrices de incidencia previa y posterior y el vector de marcado inicial de una red concreta, este algoritmo gestiona el movimiento de las marcas a lo largo de los lugares, es decir, la habilitación de transiciones, su disparo, y el cálculo del nuevo marcado. Se encuentra codificado en **ST** por sencillez en lo que se refiere a operaciones matriciales y en su transferencia a dispositivos de otros fabricantes.

El flujo de ejecución de bloques de programa es el que sigue:

1. Inicialización: se ejecuta una vez, y previamente a que corra periódicamente el ciclo de scan. Consiste en asignar al marcado en curso el valor del marcado inicial, así como calcular la matriz de incidencia a partir de la previa y posterior.

2. Sensibilización: se determinan las transiciones sensibilizadas por el marcado actual. Se compara el vector de marcado con cada una de las columnas de la matriz de incidencia previa para comprobar si el marcado sensibiliza la transición correspondiente. En caso afirmativo, se anota esta circunstancia en una cadena de transiciones sensibilizadas. Este bloque no es ejecutado otra vez hasta que se produzca algún cambio en el marcado, es decir, hasta el disparo de alguna transición.
3. Disparables: se comprueba si las transiciones actualmente sensibilizadas son disparables. Según la interpretación de la red, se evalúa el cumplimiento de las condiciones de disparo asociadas a las transiciones, listando en otra cadena si las transiciones son disparables.
4. Toma de decisión: en este bloque se implementa una regla. Nótese que una transición disparable no tiene por qué ser disparada al instante, esto puede posponerse o incluso no suceder, si el disparo de otra transición la convierte en no sensibilizada ni disparable. Esta propiedad de las PNs es particularmente interesante en el modelado de problemas de secuenciación: el control del orden en la secuencia de disparo de las transiciones. Se incorpora una función para decidir la transición a disparar entre aquellas que se encuentren sensibilizadas y que puedan hallarse en conflicto. En el caso objeto de estudio se ha aplicado la siguiente regla de decisión: la última de aquellas presentes en la lista de disparables es la que se disparará. Tras dicha elección y el consiguiente disparo, se podrá actualizar el marcado de la red.
5. Actualización del marcado: consiste en una operación de cálculo, concretamente la del nuevo marcado como consecuencia del disparo de la transición seleccionada entre las disparables. Es el resultado de la suma entre la columna correspondiente a la transición disparada en la matriz de incidencia y el vector de marcado actual.
6. Acciones: se actualizan las acciones relativas a los lugares, según el nuevo marcado y de acuerdo con lo determinado por la interpretación.

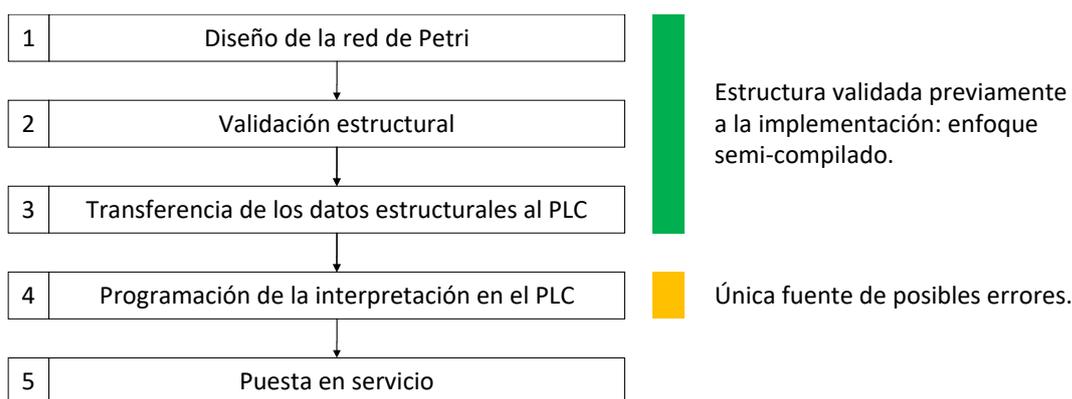


FIGURA 6.10: Fases de la metodología propuesta para la implementación de una red de Petri en un PLC.

6.2.2.4.2 Metodología de implementación

En lo que respecta a la metodología de implementación de una PN en un PLC a partir del algoritmo descrito, consta de las fases de la Figura 6.10. De ellas, sólo una puede

ser fuente de errores en el código: aquella en la que se programa la interpretación de la red. A falta de la adición manual de las mencionadas líneas y su validación, el resto del código llega a esta fase fijo y validado.

El trabajo previo a la aplicación de esta metodología, e independiente de la misma, consiste en la definición de especificaciones, caracterizando el funcionamiento deseado e identificando señales clave, como en cualquier otro proyecto de solución automatizada. Se crea el proyecto en el entorno de desarrollo; se define el hardware del PLC; se agregan los bloques de programa y se transfiere el contenido fijo e independiente de la aplicación; y se declaran e inicializan variables internas y de E/S. Además, si se va a realizar la VC del desarrollo, deberá acometerse el diseño de un DT.

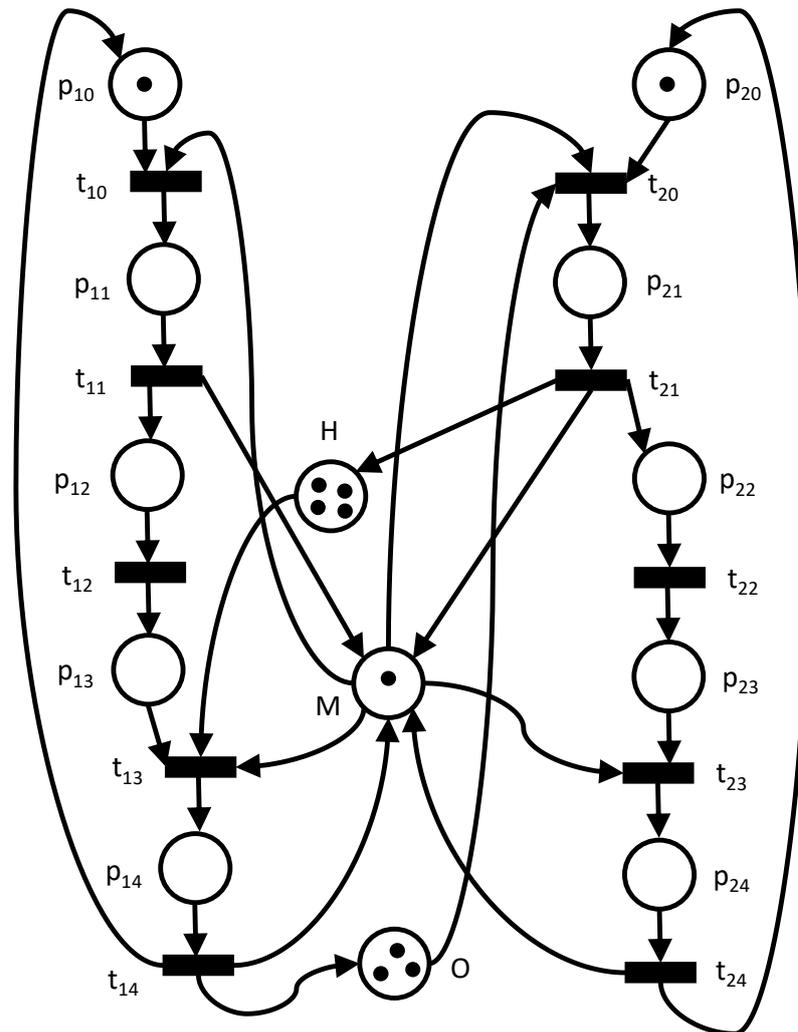


FIGURA 6.11: Estructura de la red de Petri para la célula de manufactura.

La primera fase consiste en el diseño estructural de la PN a partir de las especificaciones de funcionamiento (Figura 6.11). Se le añade la interpretación de la red, consistente en asociar condiciones y acciones a lugares y transiciones, respectivamente (Tabla 6.2).

El segundo paso se basa en una de las herramientas disponibles en el mercado con

funciones de edición de la estructura de una PN y de simulación basada en el movimiento de marcas por lugares [133]. El software PIPE2, por ejemplo, incluye opciones de análisis y validación. El cálculo de invariantes de la red permite comprobar propiedades estructurales como la limitación, la vivacidad y la ausencia de bloqueo. Asimismo, se puede generar un archivo en formato HTML con la representación matricial de la estructura, así como los resultados del proceso de validación. Nótese que no es posible una validación al 100%. Se comprueba una condición necesaria para la vivacidad, pero no garantiza totalmente esta propiedad. PIPE2 incluye la capacidad de generar (ver Figura 6.12) un Grafo de marcados accesibles o *Reachability Graph* (RG) [134]. La vivacidad puede ser verificada completamente detectando una única componente fuertemente conexa en el RG que incluya todo el conjunto de transiciones T. La Figura 6.13 muestra la representación matricial generada para la estructura de la red, con matrices de 13 filas y 10 columnas y un vector columna de marcado inicial de 13 filas que se muestra transpuesto.

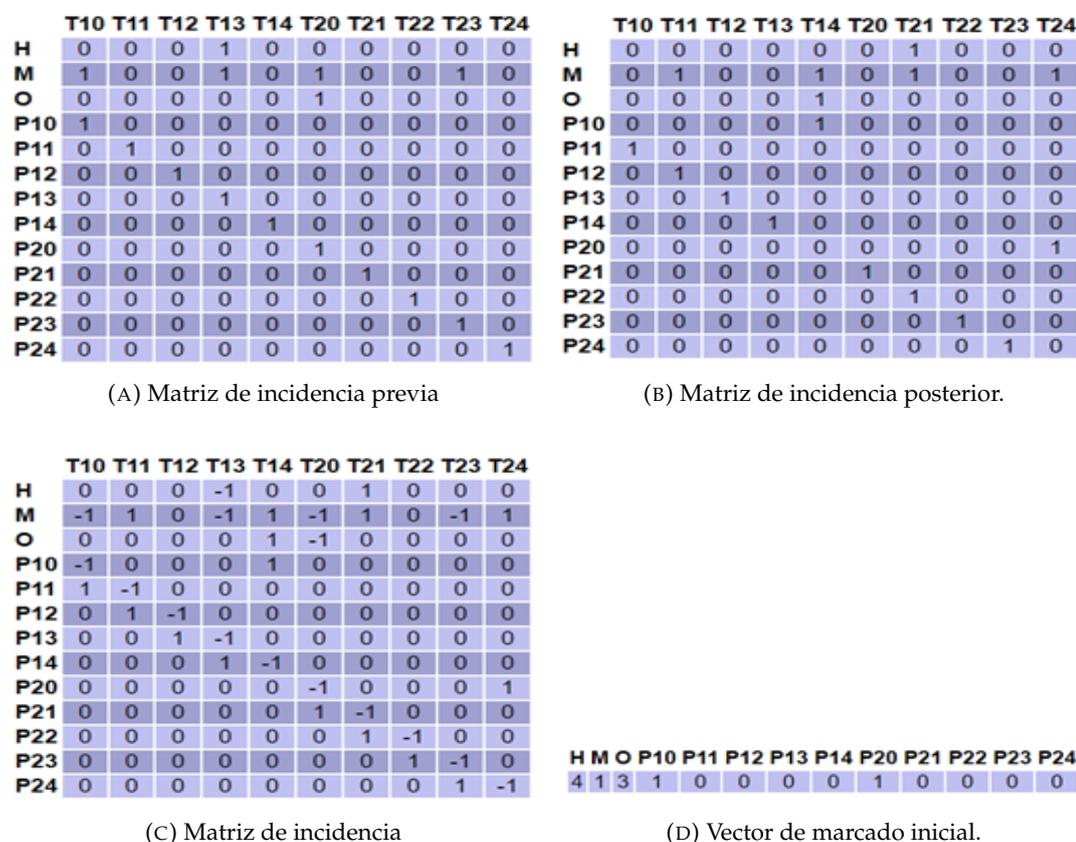


FIGURA 6.13: Representación matricial generada.

La tercera tarea consiste en la transferencia de dicha representación del archivo generado a un bloque de datos en memoria del PLC. Puede utilizarse un software propio para adaptar la información suministrada por la herramienta de edición, análisis y simulación de PNs a la sintaxis del dispositivo de control.

Posteriormente se desarrolla la única parte del código que es específica de cada caso: la evaluación del cumplimiento (o no) de las condiciones asociadas a las transiciones (Figura 6.14) y la actualización de las acciones asociadas a los lugares, según la interpretación asignada a la red.

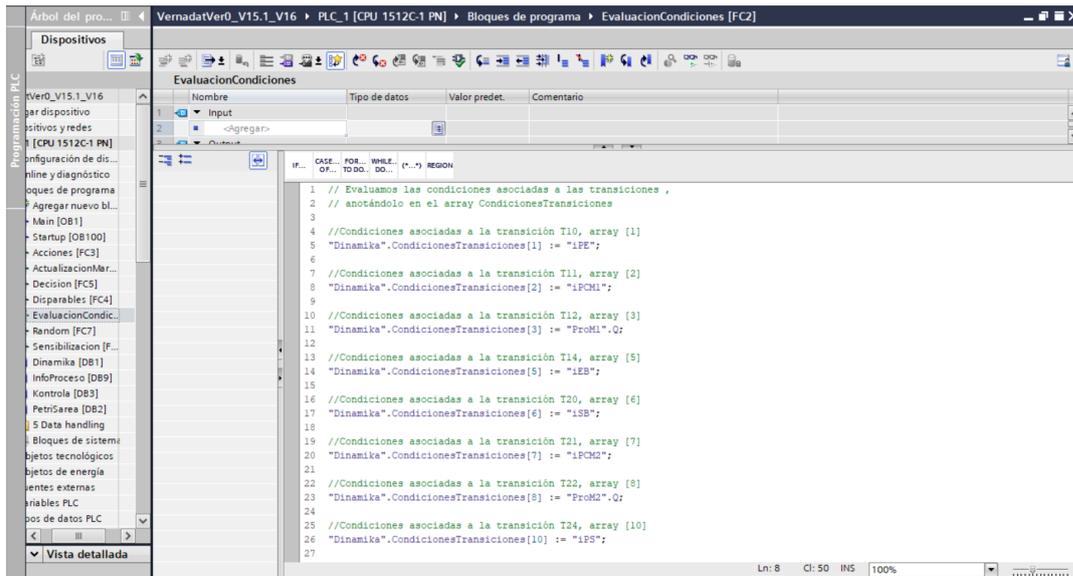


FIGURA 6.14: Detalle del código de la interpretación de una red de Petri: evaluación de condiciones.

Finalmente se verifica el funcionamiento del sistema automatizado, en cumplimiento de las especificaciones de partida.

6.2.2.5. Cierre del experimento

Con objeto de comprobar si el enfoque propuesto consume más recursos que otros similares pero de carácter interpretado se ha medido el período de ciclo de scan para cada implementación en los entornos de desarrollo de cada uno de los PLCs utilizados. Se han ejecutado más de 100000 ciclos de scan en cada prueba. Sysmac Studio de Omron facilita valores promedio (Figura 6.15). Por contra, TIA Portal de Siemens da el valor instantáneo (Figura 6.16). En este último caso se han observado 10 muestras con 10 segundos de intervalo entre ellas, y se ha calculado el promedio.

Además, y tras la primera implementación en un dispositivo de Siemens, se ha transferido el código a otro de Omron, valorando el equipo a cargo del desarrollo la portabilidad del mismo.

6.2.3. Resultados

En cuanto al período de ciclo de scan requerido por cada implementación, la Tabla 6.3 resume los resultados obtenidos. Se utiliza esta leyenda:

- **PN**: la propuesta semi-compilada, y desarrollada en **ST**.
- **ST**: implementación que procesa estructura e interpretación de una **PN**, a partir de estructuras IF THEN en **ST**.
- **LD**: desarrollo que procesa estructura e interpretación de una **PN**, a partir de la activación y desactivación (set-reset) de variables booleanas en **LDs**.
- **SFC**: implementación basada en **SFC**, método gráfico de modelado y descripción, adecuado para **GRAFSET** y disponible en el entorno TIA Portal de Siemens. Sysmac Studio no ofrece esta posibilidad.

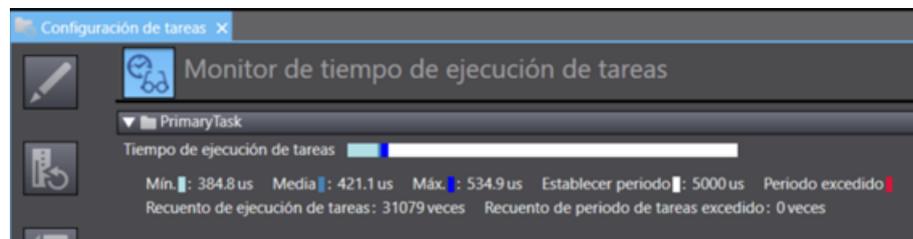


FIGURA 6.15: Medición del período de ejecución en Sysmac Studio.



FIGURA 6.16: Medición del período de ejecución en TIA Portal.

TABLA 6.3: Períodos de ejecución para las CPUs utilizadas y las implementaciones realizadas.

Implementación	CPU: Siemens 1521C-1PN	CPU: Omron NX102-1020
PN	1.029 ms	0.285 ms
ST	1.035 ms	0.276 ms
LD	1.130 ms	0.283 ms
SFC	1.361 ms	-

Más allá de la diferencia de potencial y prestaciones de los dispositivos utilizados, no se aprecian diferencias en el consumo de recursos. Si se consideran términos relativos, las implementaciones en **ST** son las más rápidas y en **SFC** las más lentas. Sin tener en cuenta esta última opción, el resto, entre las que encuentra el desarrollo objeto de estudio, ofrece un rendimiento muy similar.

En lo referente a la portabilidad de la implementación basada en **PNs** entre plataformas de distintos fabricantes, la transferencia del código en **ST** de los bloques de programa fijos es directa. La diferencia radica en las variables. La copia es directa en entornos como *COntroller DEvelopment SYStem* (**CODESYS**), pero en TIA Portal y Sysmac Studio el proceso de edición de estas va identificando sus nombres, como en cualquier otra aplicación.

6.2.4. Discusión

De los resultados del experimento se desprende que el período de ciclo de scan requerido por el enfoque propuesto no difiere del correspondiente al resto de implementaciones, enfoques similares pero con la gestión del marcado y de la interpretación integrada en en el propio código. Asimismo, su portabilidad a otro **PLC** es sencilla y rápida. El código de los bloques de programa fijos es transferido directamente. En cuanto a las variables en memoria de datos, se requiere declarar e inicializar aquellas necesarias para el funcionamiento del algoritmo. En cambio, las matrices que representan la estructura de la red son generadas en un software de diseño y simulación de **PNs**, y se adaptan mediante un software propio a la sintaxis de definición e inicialización de variables del **PLC** destino. En consecuencia, dichos resultados dan respuesta a las **RQs** de la manera que sigue:

RQ02

*¿Es el enfoque semi-compilado y basado en **PNs** más sencillo de programar que los ya existentes para **DEDSs** en **PLCs**?*

Si. La única parte que requiere desarrollo de código es la relativa a la interpretación de la red, es decir, la asignación de condiciones a transiciones y acciones a lugares. El resto es fijo e independiente de la aplicación, ya que gestiona la estructura de la red, que se trata de forma matricial en memoria de datos y viene previamente testada a partir de reglas de validación formal.

RQ03

*¿Requiere el enfoque objeto de estudio más recursos del **PLC** en términos de período de ciclo de scan?*

No. Se ha realizado una comparativa para la misma solución automatizada desarrollada a partir de diferentes enfoques.

RQ04

¿Es el enfoque objeto de estudio independiente de plataforma y sencillo de portar, tanto código como datos?

Si. Pueden requerirse pequeñas adaptaciones en la declaración de variables, pero el algoritmo, desarrollado en texto estructurado, es fácilmente transferible entre plataformas distintas.

RQ05

¿Es el DT una herramienta que facilita la validación de desarrollos genéricos contra sistemas industriales no disponibles físicamente?

Si. Hace posible testar aplicaciones contra procesos específicos en los que ponerlas a prueba, por ejemplo, por tratarse de ejemplos típicos o recomendados en la literatura especializada.

Este trabajo ha tenido por objetivo experimentar con una nueva manera de implementar **DEDSs** en **PLCs** de manera sencilla y a partir del enfoque **PN**, el más amplio y con potentes reglas de validación formal. En su cumplimiento, se ha presentado una metodología para desarrollar soluciones automatizadas de manera simple, con escasa carga de trabajo de desarrollo de código y mínimos errores potenciales. Hay adiciones o modificaciones que requieren únicamente su edición, validación automática y transferencia de los nuevos datos al **PLC**. Para quienes están habituados a trabajar con **GRAFSET**, el cambio de paradigma no es traumático, ya que se necesitan habilidades de modelado similares.

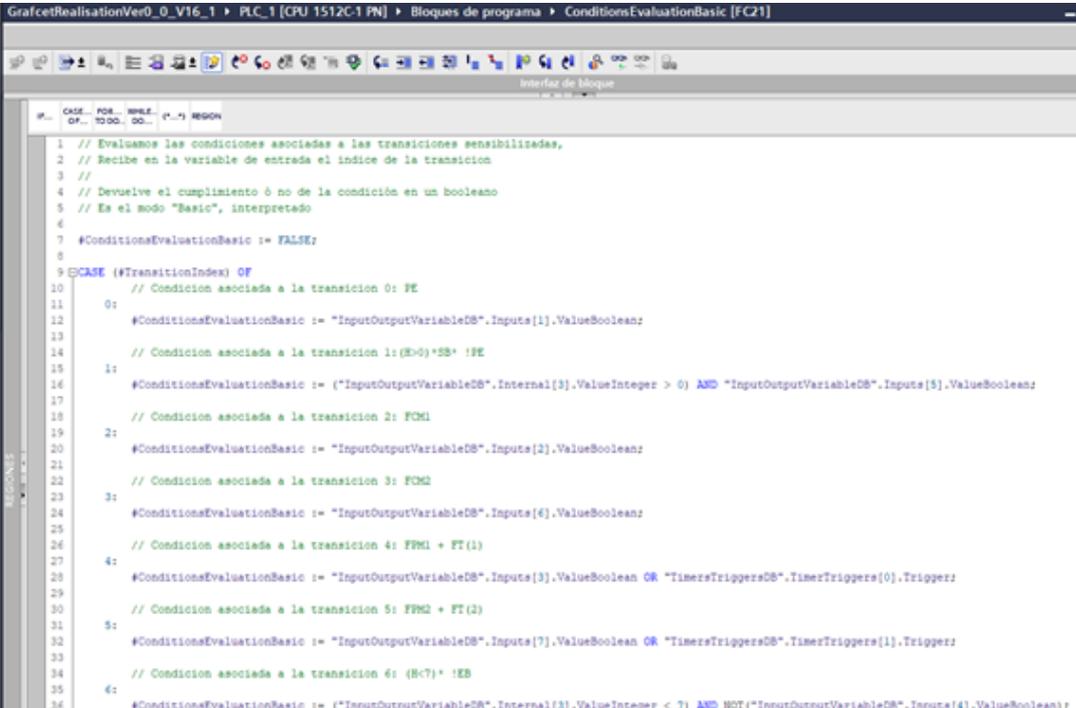
La propuesta cuenta, a priori, con una serie de debilidades, aunque alguna de ellas no debería ser tratada como tal o se subsana con sencillez:

- La programación de la interpretación, único aspecto a trabajar en el entorno de desarrollo, se realiza a criterio del personal técnico. No se dispone, de momento, de ninguna herramienta amigable o visual para ello.
- El algoritmo ocupa una estructura de proyecto completo, es decir, no es parametrizable. Su manejo sería más sencillo dándole entidad de bloque de programa o función.
- La capacidad de descripción de una **PN** es limitada. Sus matrices representativas contienen números naturales, los cuales son 0 ó 1 para **PNs** ordinarias. Extender esta propuesta a enfoques de alto nivel como una **PrT-net** o una **CPN**, donde se adjunta información a cada marca, puede aumentar la capacidad de descripción (complicando el proceso de validación) y reducir el tamaño de las redes.
- Cada entorno de desarrollo puede marcar su propio límite en lo que se refiere a las dimensiones de las matrices. En caso de necesitar superarlo, se puede recurrir a un **tipo definido por el usuario** o *User-Defined Type (UDT)*, por lo que esta posible debilidad no debe ser considerada como tal.
- Similar a la anterior, la metodología ha sido ideada para gestionar una única red. Para varias **PNs**, asociadas a sus respectivos subsistemas, bastaría considerar la unión lógica de todas ellas.

6.3. Trabajo relacionado

La evolución del trabajo, así como sus resultados, ha sido divulgada a través de un artículo de conferencia [135] y un artículo de revista [136] (ver sección 8.2.1). En el primero de ellos se introduce al lector a una primera versión de la metodología propuesta a través de su aplicación en una estación [137] de Festo Didactic, apoyada por su DT en CIROS Education 6.2. La segunda publicación está alineada a lo descrito en este capítulo. Cabe reseñar, además, el interés hacia la idea detectado en empresas fabricantes de maquinaria automatizada.

Por otro lado, al cierre de esta tesis se está sometiendo a esta metodología a una evolución que la mejora (Figura 6.17), a partir de la debilidad identificada (Figura 6.14) en la programación de la interpretación de la PN. De la codificación directa en sendos bloques de programa se pasa al uso de cadenas de caracteres que son interpretadas en tiempo de ejecución según el marcado actual. Se procederá a divulgar tanto este trabajo como sus resultados en el momento oportuno.



```

1 // Evaluamos las condiciones asociadas a las transiciones sensibilizadas,
2 // Recibe en la variable de entrada el índice de la transición
3 //
4 // Devuelve el cumplimiento ó no de la condición en un booleano
5 // Es el modo "Basic", interpretado
6
7 #ConditionsEvaluationBasic := FALSE;
8
9 CASE (#TransitionIndex) OF
10 // Condición asociada a la transición 0: FE
11 0:
12   #ConditionsEvaluationBasic := "InputOutputVariableDB".Inputs[1].ValueBoolean;
13
14 // Condición asociada a la transición 1: (E=0)*SB* !FE
15 1:
16   #ConditionsEvaluationBasic := ("InputOutputVariableDB".Internal[3].ValueInteger > 0) AND "InputOutputVariableDB".Inputs[5].ValueBoolean;
17
18 // Condición asociada a la transición 2: FCM1
19 2:
20   #ConditionsEvaluationBasic := "InputOutputVariableDB".Inputs[2].ValueBoolean;
21
22 // Condición asociada a la transición 3: FCM2
23 3:
24   #ConditionsEvaluationBasic := "InputOutputVariableDB".Inputs[4].ValueBoolean;
25
26 // Condición asociada a la transición 4: FFM1 + FT(1)
27 4:
28   #ConditionsEvaluationBasic := "InputOutputVariableDB".Inputs[3].ValueBoolean OR "TimersTriggersDB".TimerTriggers[0].Trigger;
29
30 // Condición asociada a la transición 5: FFM2 + FT(2)
31 5:
32   #ConditionsEvaluationBasic := "InputOutputVariableDB".Inputs[7].ValueBoolean OR "TimersTriggersDB".TimerTriggers[1].Trigger;
33
34 // Condición asociada a la transición 6: (B=7)* !EB
35 6:
36   #ConditionsEvaluationBasic := ("InputOutputVariableDB".Internal[3].ValueInteger < 7) AND NOT("InputOutputVariableDB".Inputs[4].ValueBoolean);

```

FIGURA 6.17: Detalle de la mejora introducida en el código de la interpretación de una red de Petri: evaluación de condiciones.

6.4. Conclusión y trabajo futuro

Se ha propuesto una metodología novedosa para la implementación de PNs en PLCs, independiente de aplicación y plataforma, en la que la mayor parte del código se valida a partir del análisis estructural den PNs antes de su implementación. La interpretación es la única parte programada y susceptible de contener errores. Para su validación se ha diseñado un caso de uso basado en un sistema de fabricación flexible, por sus características, en la literatura especializada en esta herramienta de modelado de DEDSs.

En comparación con trabajos previos en el área de interés, las líneas maestras de este experimento son:

- Se diseña y analiza formalmente una **PN**, y las matrices representativas de su estructura son obtenidas mediante una herramienta software de licencia gratuita. Se transfieren dichas matrices a memoria de datos del dispositivo de control, sea del proveedor que sea, previamente validadas.
- A partir de la citada estructura de datos se gestiona la evolución del marcado de la red mediante bloques de programa fijos e independientes tanto del **PLC** como de la **PN**. Por ello, este enfoque puede ser considerado como semi-compilado. El hecho de que la parte fija esté programada en **ST** la lleva a poder ser considerada como estándar.
- Sólo hay que programar el código relativo a la interpretación de la red. La asociación de condiciones a transiciones y acciones a lugares puede realizarse en cualquier lenguaje soportado por el correspondiente entorno de desarrollo.

La validez de la metodología presentada se sustenta en los resultados obtenidos en términos de período de ciclo de scan requerido y portabilidad, así como su implementación en un sistema industrial. Dicho sistema es referido en la literatura como ejemplo típico de aplicación de **PNs**, y exigente para un enfoque basado en **GRAF-CET** por el uso de marcado múltiple que requiere código extra al no formar parte de la estructura del modelo.

La emulación ha hecho posible trabajar contra un proceso industrial específico no disponible físicamente y en el que se quería poner a prueba el desarrollo. Sin un **DT** el testado se basaría en el convencional forzado de entradas y observación de salidas del **PLC**, más laborioso, lento y donde puede dejarse accidentalmente sin reproducir algún escenario o situación.

Se estima que este trabajo puede ser ampliado a partir de las líneas de trabajo futuro identificadas, y basadas en las debilidades ya comentadas:

- Reducir aún más la carga de trabajo de codificación haciendo más sistemática la adición de interpretación a la **PN** puede ser un reto interesante. Existe una importante investigación sobre la generación automática de código.
- La parametrización de un bloque de programa que encapsule el algoritmo puede facilitar su manejo.
- Aumentar la capacidad de descripción de una **PN** a partir de un enfoque de alto nivel en el que se apliquen técnicas orientadas a objetos y/o incorporando atributos a marcas, lugares (acciones) y transiciones (condiciones), reduciendo las dimensiones del modelo. Los **UDTs** podrían ayudar en la programación del **PLC**.
- También puede ser interesante implementar el modelo en MatLab, Java y/o ADA, para comparar tanto el coste de desarrollo como su rendimiento.
- **PNs** distribuidas, en las que se utiliza la gestión de una red maestra para operar sobre las demás, a través de protocolos de comunicación abiertos como **OPC UA** o *Message Queuing Telemetry Transport (MQTT)*.

Capítulo 7

Diagnóstico de actividad en sistemas industriales automatizados.

En un entorno industrial automatizado se dispone de un gran volumen de datos y señales, relativo tanto a sensores y actuadores instalados en la maquinaria como a la interacción con operarios y usuarios. El diagnóstico de la actividad puede tener múltiples aplicaciones desde el punto de vista del aprendizaje o la formación de personal novel, así como en términos de evaluación de procesos. En este capítulo se presenta una metodología de implementación de un **IS** a partir de la integración, en un marco genérico de diagnóstico contextual, de cualquier sistema interactivo con conectividad **OPC UA** y un **DT** del sistema industrial. Se interpreta y diagnostica la actividad según el contexto en el que se produce, con el objetivo de validar automáticamente el funcionamiento del sistema. En el caso objeto de estudio, basado en una célula robótica controlada por **PLC** en la que se opera tanto en modo automático como en modo manual para ajustar el posicionamiento de una mesa lineal, se analizan el orden de los pasos en la secuencia, la duración de los mismos y la duración total de la secuencia.

7.1. Introducción

El trabajo que se describe a continuación ha sido llevado a cabo en la actividad de I+T, concretamente en el testado del potencial de las tecnologías de digitalización más allá de la validación del código de sistemas de control.

La digitalización de entornos industriales inteligentes en el contexto de la Industria 4.0 trae consigo la adquisición de grandes volúmenes de datos con el fin de supervisar y controlar diferentes aspectos de los procesos. El uso de estos datos de proceso está permitiendo desarrollar sistemas con mayor nivel de autonomía para garantizar un funcionamiento seguro, el control de la calidad del producto y la evaluación de la integridad del equipamiento. Para dotar de mayor autonomía o conciencia acerca del contexto a un sistema, este debe ser capaz de percibir e interpretar su entorno y adaptar su comportamiento en consecuencia. Este experimento tiene por fin el diagnóstico de actividad de operario, estudiante o proceso industrial a partir de la detección del contexto por medio de situaciones.



FIGURA 7.1: Simulador de camión.

Fuente: [141].

La literatura se hace eco de trabajos relacionados con la conciencia de contexto, con aplicaciones en áreas industriales como la manipulación avanzada de materiales [138], o la monitorización, donde [139] integra los datos de contexto y la información existente de la planta para proporcionar dinámicamente sólo la información más relevante. Además, una herramienta desarrollada por [140] permite una (re)configuración consciente del contexto de los dispositivos de campo de un sistema.

La gran mayoría de los procesos industriales tienen definidos procedimientos que caracterizan un funcionamiento adecuado de los mismos. Para cada una de las funciones y tareas relacionadas con un proceso (funcionamiento automático normal, ajuste manual, mantenimiento, etc.), diferentes procedimientos definen secuencias de acciones que deben cumplirse para realizar correctamente la operación.

En este capítulo se propone la integración de sistemas interactivos con conectividad **OPC UA** en un marco ya existente, para el diagnóstico de procesos industriales automatizados interpretando, a partir de los datos del mismo, el contexto y el cumplimiento de los procedimientos asociados a las tareas u operaciones. **OPC UA**, basado en el principio cliente-servidor e independiente de fabricante, es el protocolo de comunicación industrial idóneo para ello. Permite la lectura y escritura transparente de variables en la memoria del controlador, normalmente un **PLC**. Los dispositivos de control de última generación y las herramientas de emulación de procesos incluyen conectividad **OPC UA**, considerado como el estándar de la Industria 4.0. Los **DTs** de sistemas industriales automatizados intercambian señales con dispositivos de control a través del citado estándar y pueden contribuir a la reproducción de múltiples escenarios en aplicaciones exclusivamente virtuales o para la configuración y validación previa a la puesta en servicio de aplicaciones que se ejecutarán posteriormente en el sistema físico, sin comprometer la seguridad, la operativa ni la producción en este último.

7.1.1. Antecedentes y enmarque

Bajo la influencia de lo que se conoce como aprender haciendo, en 2012 se propuso una metodología para la integración de sistemas interactivos, en la que se dotó a los simuladores de capacidades de evaluación. A partir de modelos de interpretación y

diagnóstico según el [modelado basado en restricciones](#) o *Constraint-Based Modeling (CBM)* se acometieron varios desarrollos. El primero de ellos tuvo por sistema interactivo un simulador de camión (Figura 7.1). Se implementó un sistema de apoyo al aprendizaje, para conocer lo que se estaba realizando en cada momento, y dar una formación de bajo coste y eficiente a profesionales de la conducción. La segunda plataforma apoyaba la formación de personas con discapacidad en el manejo de herramientas de jardinería. El tercer desarrollo se basó en habilidades físicas en el deporte del tenis.

El punto de partida de este trabajo ha sido, en consecuencia, un marco genérico de diagnóstico que ha sido aplicado en manejo manual de herramientas y en destreza física, con simuladores a modo de sistema interactivo. Por una parte, se ha dispuesto del núcleo de ejecución y la metodología para integrar sistemas interactivos y desarrollar *ISs*. Por otra, se ha considerado que la capa de diagnóstico puede ser de utilidad para detectar averías en un sistema industrial automatizado. Siendo *OPC UA* un protocolo de uso extendido tanto en dispositivos de control como en herramientas de emulación, se ha planteado el desarrollo de una metodología para implementar un *IS* mediante la conexión, a través de dicho estándar, de cualquier sistema interactivo, no necesariamente un simulador, sino por ejemplo un *PLC* industrial.

A partir de la integración de dicho dispositivo de control en la plataforma genérica, los datos y señales disponibles (sensores, actuadores, interacción con personas) pueden apoyar la interpretación y diagnóstico de la actividad en los procesos industriales automatizados en función del contexto en el que se producen. Las aplicaciones pueden ser múltiples, como (i) la verificación de la correcta realización de una tarea contra el procedimiento ideal predefinido, (ii) la extracción del procedimiento ideal basado en un operador experto, y (iii) la formación y orientación de personal novel.

7.1.2. Fundamentos del marco

Automatic Diagnosis of Activity and Processes in auTomation environments (ADAPT) es un marco basado en el metamodelo ULISES y su núcleo de ejecución [142], el cual se alimenta de tres modelos, el de observación, el de interpretación y el de diagnóstico. Dicho metamodelo es independiente del dominio, y está inspirado en el proceso cognitivo llevado a cabo por expertos o tutores del entorno, por ejemplo en escenarios educativos, mientras supervisan o tutelan a un alumno durante un proceso. A partir de la percepción de hechos u observaciones, los expertos interpretan esas observaciones para identificar las acciones. Asignan inconscientemente un significado a sus observaciones en el contexto del dominio. Una vez procesada esta información, realizan un diagnóstico consistente en detectar y/o corregir errores.

Hay, por tanto, (i) un experto que sabe cómo se solventa el proceso, (ii) alguien en proceso de aprendizaje, y (iii) la figura de diseñador instruccional, que se encarga de la inserción de modelos de interpretación y diagnóstico al sistema. Definir la información necesaria para los modelos de observación e interpretación no es una tarea obvia. Lo que al principio se realizaba editando bases de datos culminó en la creación de una herramienta de autoría [143] llamada PATH [144], para que cualquier diseñador instruccional pudiera insertar el conocimiento del dominio a ULISES sin tener que abrir ficheros y editar los modelos. La herramienta PATH permite:

- La edición de modelos de conocimiento (modelos de interpretación y de diagnóstico) por parte del diseñador instruccional.
- La monitorización de señales del sistema interactivo.

- La visualización, interpretación y diagnóstico en tiempo real del funcionamiento de los modelos a partir de la información generada por ULISES y publicada mediante agentes.
- La generación de informes finales.

7.1.2.1. Niveles lógicos

El núcleo de ejecución de ULISES está compuesto por los subsistemas de (i) observación, (ii) interpretación y (iii) diagnóstico y define un metamodelo dividido en tres niveles lógicos que representan genéricamente el mismo proceso: (i) observación, (ii) interpretación, y (iii) diagnóstico, el cual se basa tanto en las observaciones como en la interpretaciones.

7.1.2.1.1 Nivel de observación

Este nivel contiene los elementos necesarios para especificar los hechos observables que interesan desde un punto de vista educativo o de diagnóstico de procesos. Estos elementos se denominan observaciones y representan eventos que tienen lugar durante un intervalo de tiempo en el sistema interactivo, el cual únicamente emite flujos de datos. Para el caso concreto de *ADAPT*, el sistema interactivo a integrar es un servidor *OPC UA*. El objetivo del nivel de observación es especificar cómo el subsistema de observación tiene que transformar dichos flujos de datos en observaciones, lo cual requiere un laborioso trabajo de programación. Si se recibiera información de sensores que lleva adosados una persona, se podría generar una observación de la rodilla derecha. Una observación puede tener propiedades, como por ejemplo la velocidad de elevación y el movimiento en el eje vertical, para la observación de la elevación de la rodilla derecha. Las observaciones son las primitivas para los niveles de interpretación y diagnóstico. Por lo tanto, el trabajo del subsistema de observación es fusionar las entradas del sistema interactivo para transformarlas en observaciones y actualizarlas de forma sincronizada para los dos subsistemas superiores: el de interpretación y el de diagnóstico.

7.1.2.1.2 Nivel de interpretación

El nivel de interpretación describe de forma genérica cómo reconocer los sistemas interactivos con tal precisión que el subsistema de diagnóstico sea capaz de determinar si las acciones son correctas o incorrectas, y sus razones. Para ello es imprescindible conocer el contexto en el que se producen las acciones. El nivel de interpretación define dos elementos, pasos y situaciones, que se generan a partir de las observaciones utilizando el paradigma de modelado de restricciones [145]:

- Paso: representa una acción que tiene lugar en un intervalo de tiempo y que será diagnosticada. Un paso define cómo el núcleo de ejecución de ULISES analizará las observaciones para interpretar cuándo se está realizando un paso. Teniendo en cuenta que las observaciones son duraderas, se describen las relaciones temporales entre ellas. Es importante señalar que un paso no debe ser interpretado en base a la corrección o incorrección de una acción. Lo que debe hacer este nivel es interpretar las acciones cuando se realizan, independientemente de su corrección. Por ejemplo, si un sistema interactivo es un simulador de conducción, un paso puede ser un cambio de carril.

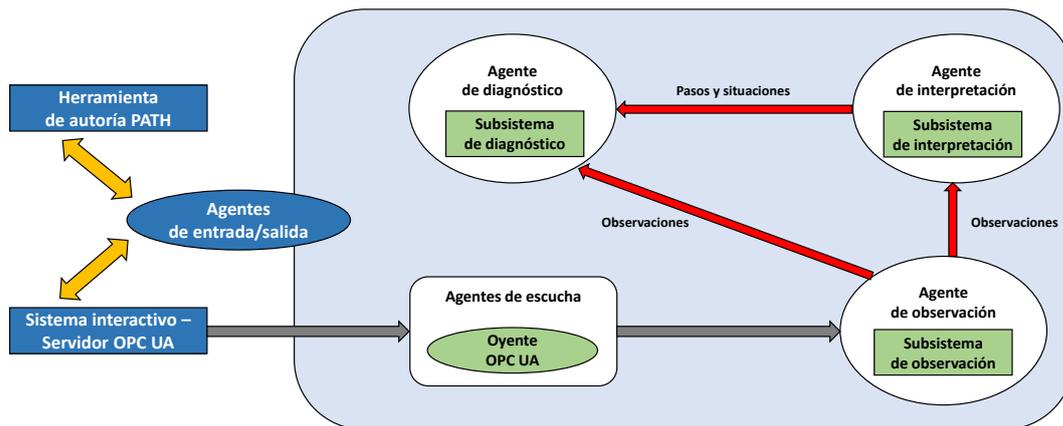


FIGURA 7.2: Interacción entre los agentes de ADAPT.

- **Situación:** se compone de pasos y representa el contexto en el que se ejecutan las acciones. En el caso del simulador de conducción, si se quiere evaluar un cambio de carril al adelantar, es necesario saber si el conductor está en una situación de adelantamiento. La acción no sería relevante en un contexto de abandono de carril por aproximación a una salida.

7.1.2.1.3 Nivel de diagnóstico

Tiene por objetivo gestionar los elementos que permitan generar un diagnóstico adecuado de la actividad. En ULISES, este nivel permite integrar diferentes componentes de diagnóstico como los módulos de diagnóstico basados en restricciones y comunicarse con sistemas de diagnóstico externos como por ejemplo DETECTive [146]. ULISES proporciona a los componentes educativos o de evaluación de procesos del sistema construido resultados de diagnóstico en tiempo real o resultados de evaluación final según sea necesario. Los resultados de diagnóstico contienen información a diferentes niveles: la corrección de las situaciones, la corrección de cada paso dentro de cada situación y otra información como las condiciones satisfechas y no satisfechas dentro de los pasos.

7.1.2.2. Núcleo de ejecución de ULISES

El núcleo de ejecución de ULISES está diseñado a partir de una arquitectura multiagente. El concepto de **sistema multiagente o Multi-Agent System (MAS)** tiene su origen en la inteligencia artificial distribuida, pero se ha generalizado en la automatización en la última década [147] [148]. En cuanto a la comunicación entre agentes, ULISES se basa en el estándar FIPA de interoperabilidad [149]. Proporciona la comunicación entre cualquier servicio externo y las aplicaciones y los tres subsistemas que componen el núcleo de ejecución. Cada subsistema de ULISES se comunica con los demás principalmente a través de los protocolos estándar de suscripción, solicitud y consulta. Por lo tanto, cada agente puede ofrecer y solicitar la información que necesita justo en el momento requerido. La Figura 7.2 muestra la relación entre los agentes de ULISES y el resto del sistema inteligente:

- **Agentes de entrada-salida:** permiten que el núcleo de ejecución de ULISES se comunique con interfaces de usuario u otros servicios como la herramienta de autoría PATH.

- Agente de observación: encapsula el subsistema de observación y se encarga de recoger los datos del servidor OPC UA a través de los agentes de entrada-salida y transformarlos en observaciones para los subsistemas de interpretación y diagnóstico.
- Agentes de escucha: mantienen la comunicación entre los sistemas. En el caso de ADAPT, servidor(es) OPC UA y el agente de observación. Hay que tener en cuenta que puede haber varios agentes en caso de sistemas distribuidos.
- Agente de interpretación: recibe las observaciones a las que está suscrito, y luego genera los pasos y situaciones que detecta utilizando el modelo de interpretación.
- Agente de diagnóstico: este agente coordina el diagnóstico de los pasos y situaciones recibidos del agente intérprete. El proceso de diagnóstico puede llevarse a cabo internamente, con componentes de diagnóstico específicos integrados con el subsistema de diagnóstico a través de complementos, o externamente. Por ejemplo, en el caso del sistema DETECTive ya mencionado, los complementos de envoltura gestionan su comunicación con el subsistema de diagnóstico.

7.1.3. Definición del problema

A partir del contexto descrito, el trabajo que se describe a continuación tiene por objetivo dar soporte a actividades como la formación de personal de planta, la cual se lleva a cabo en el propio equipamiento industrial, con la consiguiente influencia en el funcionamiento y el ritmo productivo de este, y comprometiendo tanto su integridad como la seguridad de las personas. La verificación del cumplimiento de procedimientos que caracterizan el funcionamiento adecuado de procesos industriales se realiza mediante la supervisión de personal experto.

Se pretende realizar dicha verificación de manera automática en un DT del sistema industrial para aplicaciones de entrenamiento, de manera que no se interfiere en la operativa de la maquinaria y se evitan riesgos, o en el propio sistema físico para el diagnóstico, sin supervisión humana, de operaciones automáticas o manuales. Para ello se propone la integración de cualquier sistema interactivo con conectividad OPC UA, como un PLC, en un marco de diagnóstico ya existente.

7.2. Evaluación empírica

En esta sección se exponen aspectos relativos a la implementación realizada. Se enuncian las RQs a las que se ha pretendido dar respuesta. Posteriormente se caracteriza el ensayo realizado y se presentan los resultados obtenidos, para finalizar con la discusión de los mismos.

7.2.1. Preguntas de investigación

A partir de lo descrito y de lo extraído de la literatura, se han enunciado estas RQs:

RQ06

¿Puede integrarse un sistema interactivo con conectividad OPC UA en un marco genérico de diagnóstico contextual de actividad a partir de dotar a este último de dicha conectividad?

RQ07

¿Constituye un marco genérico de diagnóstico contextual de actividad, al que se ha integrado como sistema interactivo un PLC que controla un sistema industrial automatizado emulado a través de su DT, una herramienta virtual para entrenamiento/formación de personal inexperto?

RQ08

¿Constituye un marco genérico de diagnóstico contextual de actividad, al que se ha integrado como sistema interactivo un PLC que controla un sistema industrial automatizado, una herramienta para diagnóstico de actividad en aplicaciones industriales reales?

7.2.2. Caracterización del ensayo

Esta sección tiene por objeto detallar el experimento realizado. Se divide en los siguientes bloques: el listado de material utilizado, las características del DT del sistema industrial automatizado objeto de estudio, la descripción del trabajo llevado a cabo y las condiciones del mismo, la metodología seguida y la obtención de resultados.

7.2.2.1. Material utilizado

A continuación se listan los recursos requeridos para un desarrollo de estas características. Se especifican, si procede, las versiones de las que se ha dispuesto para esta aplicación en concreto.

- El sistema que identifica, verifica y clasifica carcasas de dron de la Figura 5.1. Se remite al lector al Capítulo 5 para más detalles.
- Un PLC con conectividad OPC UA.
Se ha hecho uso de un controlador de la serie S7-1500 de Siemens.
- Un panel HMI para la interacción con usuarios.
Concretamente un panel táctil de la serie TP700 Comfort de Siemens.
- Herramientas software:
 - Software de simulación y programación offline de robots.
En este caso, RobotStudio 2019.1 de ABB.
 - Entorno de desarrollo para dispositivos de control.

A pesar de hallarse operativos PLC y HMI, se ha recurrido a TIA Portal V15.1, para la monitorización de variables e inserción de errores de funcionamiento para su diagnóstico.

- Un sistema **SCADA** para la interacción con usuarios.
Se ha dispuesto de una aplicación desarrollada en Wonderware InTouch 2014 R2.
- Una plataforma para dotar de conectividad **OPC UA** al **SCADA**.
Se trata de KEPServerEX 6.2 con Wonderware OPC UA Client Service.
- La versión emulada del sistema de manufactura, en forma de **DT** en **S3D**, para el testado de la aplicación de diagnóstico y su ejecución en caso de que sea exclusivamente virtual
- Cliente **OPC UA** para dotar de conectividad a través de dicho estándar al marco **ADAPT**.
Se ha dispuesto de .NET Based OPC UA Client.
- El marco en si, incluyendo la metodología y el software **PATH** para crear sistemas interactivos inteligentes.

7.2.2.2. Caracterización del modelo virtual

Las características con las que cuenta el **DT** (Figura 5.3) con el que se ha realizado a cabo este experimento son detalladas en el Capítulo 5. Se remite al lector al mismo para más detalles.

7.2.2.3. Trabajo realizado

En esta sección se enmarca la implantación del marco **ADAPT** en el caso de uso. Se describe el trabajo realizado en cuanto a los modos de operación analizados, el diseño de los modelos asociados, y cómo se ha testado e integrado el marco de diagnóstico. Posteriormente se describen las observaciones, interpretaciones y diagnósticos llevados a cabo.

7.2.2.3.1 Definición del trabajo

Las tareas y los modelos de interpretación han sido definidos mediante la herramienta de autoría **PATH**, integrada en el marco de diagnóstico. Se comprueba la corrección del funcionamiento de los modelos a partir del uso, por parte de dicha herramienta, de la información generada por **ULISES** y publicada mediante agentes.

Una vez integrado el **PLC** en la plataforma, se ha testado la misma mediante el intercambio de señales entre el dispositivo de control y el **DT** del sistema de manufactura, en lo que podría ser una aplicación exclusivamente virtual para formación de personal. Una vez validada la plataforma se ha trabajado con el sistema físico, pasando el **PLC** a intercambiar señales con sensores, actuadores y el robot, ya no emulados sino físicos, a través de E/S. Esto puede constituir una aplicación de diagnóstico contextual en un entorno industrial automatizado.

Se ha operado en la célula robótica, tanto emulada en primera instancia como real posteriormente, de dos modos. Estos han sido representados en los modelos de interpretación y diagnóstico como situaciones automática y manual. El subsistema de diagnóstico debe garantizar que, en la situación automática (proceso que se ejecuta en modo automático), (i) se realicen todos los pasos definidos y (ii) se ejecuten todos los pasos con una duración definida. La duración excesiva de uno de los pasos

TABLA 7.1: Pasos y situaciones a implementar.

Situación	Operación en modo automático.
Pasos	Solicitud de permiso de acceso al recinto para situar carcasa. Dar inicio al proceso. Identificación de color y altura de la carcasa. Clasificación de la carcasa.
Situación	Ajuste de la posición lineal de la mesa.
Pasos	Con la mesa en la posición de entrada, selección de modo manual. Movimiento a posición predefinida de testado de altura. Ajuste de posición de testado de altura (puede omitirse). Validación de posición de testado de altura. Movimiento a posición predefinida de testado de color. Ajuste de posición de testado de color (puede omitirse). Validación de posición de testado de color. Movimiento a posición predefinida de salida. Ajuste de posición de salida (puede omitirse). Validación de posición de salida.

puede deberse a la degradación de algún componente o mecanismo. En el modo manual, el enfoque de **ADAPT** es educativo, diagnosticando si los usuarios siguen correctamente los pasos establecidos para completar una secuencia, y en un tiempo adecuado.

7.2.2.3.2 Qué se ha observado, interpretado y diagnosticado

A partir de las señales del proceso y de la interacción con el usuario, se han definido e implementado los pasos y situaciones de la Tabla 7.1. Las situaciones corresponden al funcionamiento normal o automático y al ajuste de posición lineal de la mesa en modo manual, en ambos casos con sus respectivas secuencias de pasos.

Las observaciones son eventos en el sistema interactivo (**PLC**) que se combinan en restricciones asociadas a pasos en situaciones. Por ejemplo las señales observadas para la operación del modo automático son la posición del selector, la solicitud de acceso, la orden de inicio y la consigna de posición lineal.

Los pasos y situaciones descritos han sido implementados en el modelo de interpretación a partir de la definición de restricciones asociadas a cada paso. Se trata de restricciones de inicio, de finalización, generales o de acción, entre otros tipos. La Figura 7.3 muestra la definición, en **PATH**, de una restricción asociada a un paso. En cada ciclo de ejecución del algoritmo del marco se resuelven todas las reglas y se determina el estado de cada paso, es decir, si comenzó, terminó, continúa, etc. (Figura 7.4). A partir del reconocimiento del contexto basado en las observaciones, se verifica el cumplimiento de restricciones para pasos y situaciones. Se trata, para la

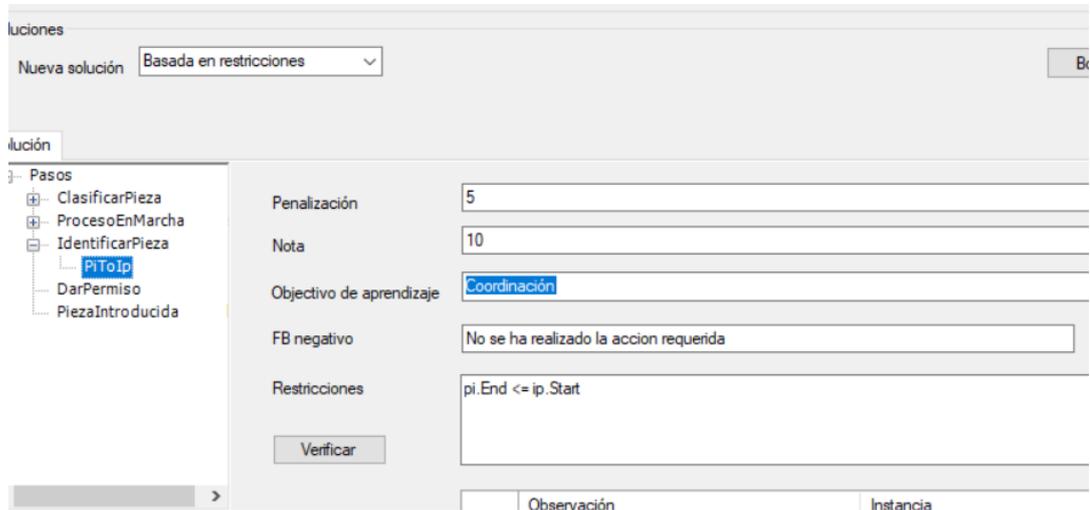


FIGURA 7.3: Definición de una restricción asociada a un paso.

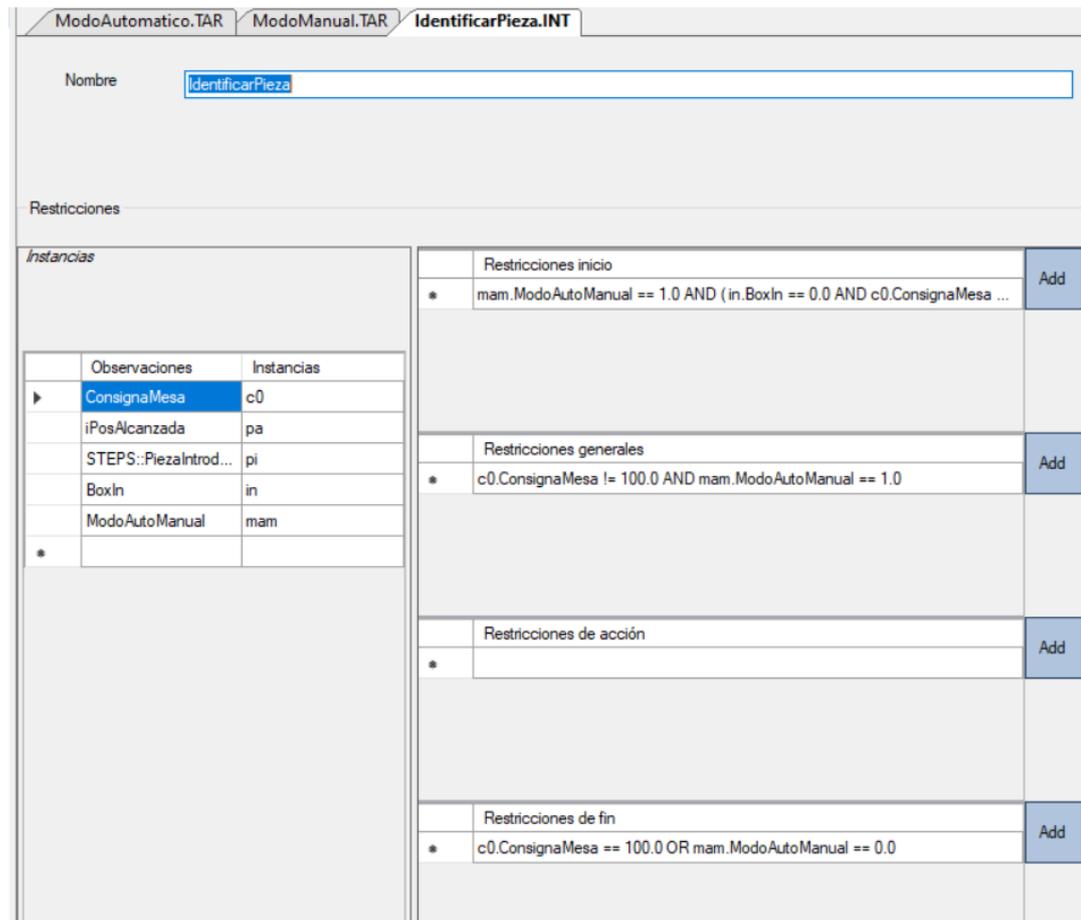


FIGURA 7.4: Restricciones asociadas a un paso.

situación relativa al modo automático, del de proceso en marcha, el de identificación de pieza, y el de clasificación de pieza.

Puede repetirse toda la secuencia tantas veces como sea necesario. Al final de la misma se podría pasar al modo automático, si así se requiere. Si se volviera al modo automático antes de finalizar el procedimiento, se consideraría erróneo. Se verifica la corrección de situaciones y pasos en términos de orden de pasos en secuencia, duración de paso y duración de secuencia. El diagnóstico de las tareas que se exceden en tiempo o de los pasos que no se ejecutan en el orden correcto está basado en la gramática de Allen [150]. El mismo paso puede ser correcto o incorrecto, según en qué situación se produzca.

7.2.2.4. Metodología

Al ser ULISES un marco genérico, el nivel de observación no puede predeterminar cuáles son las características del sistema interactivo. Cada sistema construido con este marco, como por ejemplo [ADAPT](#), necesita una integración en el nivel de observación. Esta sección describe cómo se lleva a cabo esta integración en [ADAPT](#).

7.2.2.4.1 Integración del marco a través de OPC UA

El objetivo principal del nivel de observación es proporcionar los elementos necesarios para describir cómo observar los hechos que están siendo recogidos por el servidor [OPC UA](#) y que son de interés. Esos hechos serán las primitivas para los niveles superiores, que describirán la actividad del proceso (lo cual es interesante desde el punto de vista educativo) en tiempo real. Por ejemplo, si se quiere saber si una pieza está siendo clasificada en un periodo de tiempo determinado, se deberá observar si la pieza ha llegado a la cámara y se ha realizado su detección.

Subsistema de observación

El propósito del subsistema de observación es recopilar las secuencias que provienen del servidor [OPC UA](#) y convertirlas en observaciones. Los dos elementos principales que participan en este proceso de transformación son el agente de escucha [OPC UA](#) y el agente de observación [OPC UA](#).

- Agente de escucha [OPC UA](#): comunica el marco con el servidor. Publica el comienzo y final de un nuevo ciclo y comunica el flujo de datos que ha recibido del servidor. Este agente contiene un controlador de cliente [OPC UA](#) que establece la conexión con el servidor externo y crea suscripciones en este. Estas suscripciones están basadas en las observaciones que han sido predefinidas en el modelo de observación. Una vez recogidos todos los datos, son publicados para todos los agentes que estén suscritos a esta información.
- Agente de observación [OPC UA](#): es el responsable de convertir el flujo de datos en observaciones reales. Recibe los datos necesarios a través de una suscripción al agente de escucha [OPC UA](#), y envía estos datos al observador [OPC UA](#). El observador implementa un cliente [OPC UA](#) para generar observaciones en cada ciclo. Cuando la evaluación y transformación de los flujos de datos finaliza, el agente observador notifica a los agentes intérprete y de diagnóstico que las observaciones han sido actualizadas. Estas observaciones son almacenadas en un fichero de configuración, y son definidas en función de la interpretación y el diagnóstico que se pretende realizar para el proceso.

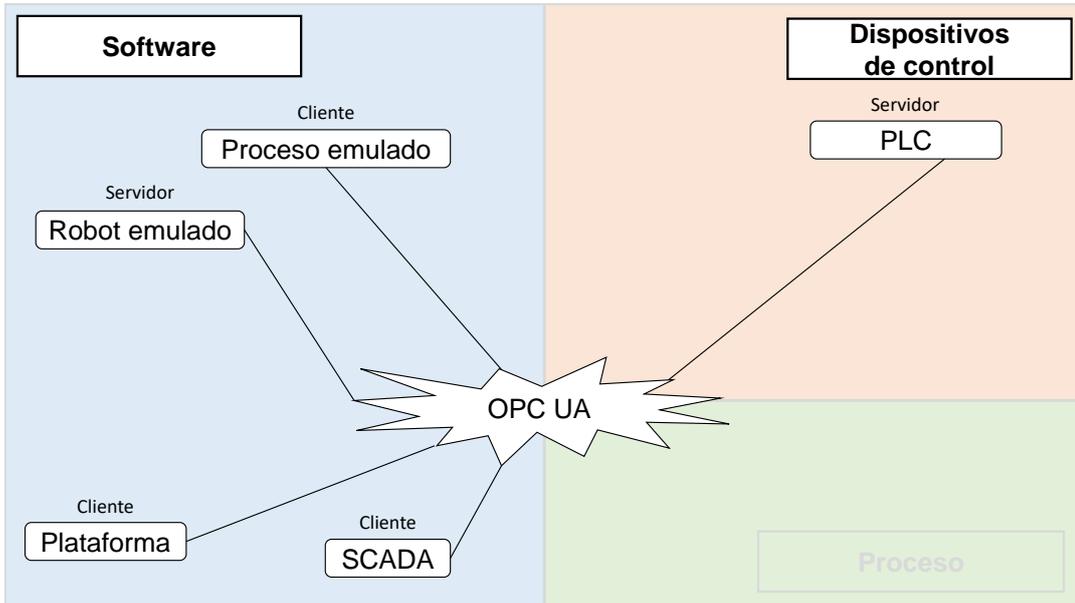


FIGURA 7.5: Configuración del sistema de control en el testado de la plataforma: posible aplicación como herramienta de entrenamiento de personal novel.

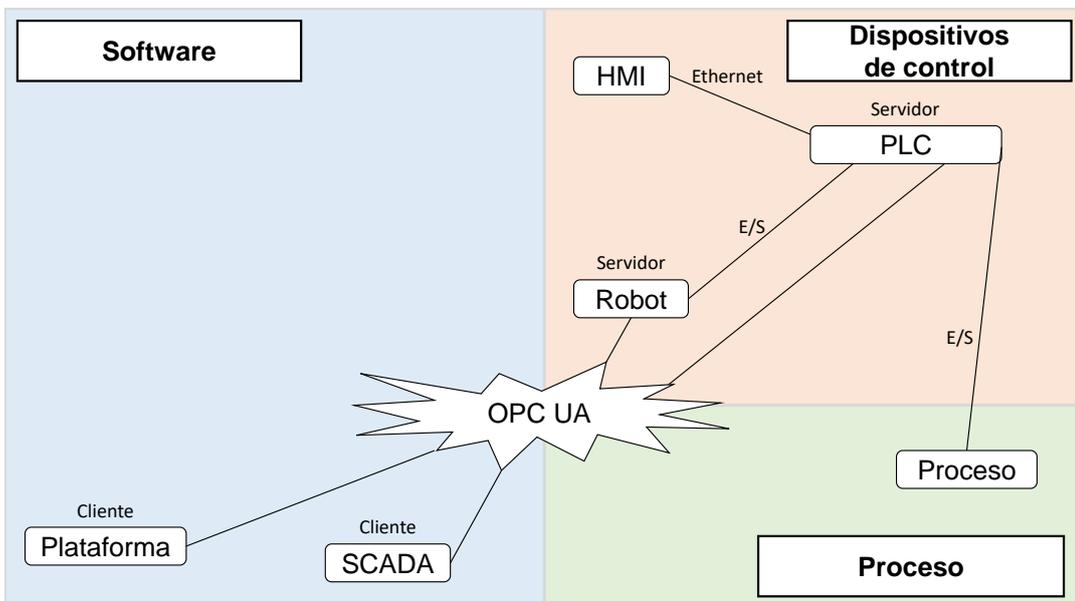


FIGURA 7.6: Configuración del sistema de control en la fase operativa del proceso físico: posible aplicación como herramienta de diagnóstico de actividad.

Subsistema de interpretación

Una vez generadas las observaciones, el subsistema de interpretación o intérprete discierne qué pasos y situaciones se están ejecutando en el sistema interactivo. Esos pasos y situaciones se modelan mediante un enfoque basado en restricciones [142]. Es decir, el usuario establece restricciones para definir esos dos elementos y en cada ciclo se evalúan esas restricciones. Si se satisfacen dichas restricciones, se crean pasos y situaciones y esta información es enviada al subsistema de diagnóstico superior.

Subsistema de diagnóstico

En cuanto al subsistema de diagnóstico, este procesa los pasos y situaciones identificados por el intérprete. Para cada paso y situación se define una solución que se diagnostica en función de lo definido en el modelo de diagnóstico. Este modelo se basa también en una técnica de modelado de restricciones. En resumen, se especifica cómo deben llevarse a cabo esos pasos y situaciones. Por ejemplo, si se ejecuta una simulación en la que interviene un ser humano, su actividad podría ser evaluada en tiempo real. Asimismo, si se está observando un proceso, se puede especificar el correcto funcionamiento de cada situación y paso del mismo. Por lo tanto, este enfoque hace que **ADAPT** sea una solución adecuada para los controladores y **DT** basados en **OPC UA**, estándar comúnmente utilizado para la conectividad entre **PLCs** (reales o emulados) con **DT** de procesos industriales, con el fin de testar software de control.

7.2.2.4.2 Configuraciones para testado e integración

En primera instancia, para la integración de dispositivos y aplicaciones que forman parte del marco mediante el estándar **OPC UA**, se ha configurado el sistema de diagnóstico según lo mostrado en la Figura 7.5. Los dispositivos de control, en este caso un **PLC** físico y un robot emulado, han sido configurados como servidores. La plataforma accede a datos del primero como cliente. Se ha probado una interacción básica con usuarios en un **SCADA** también cliente. Otro cliente, el sistema industrial emulado a través de su **DT**, ha hecho posible testar la plataforma disponiendo de señales en el **PLC** sin que procedan del proceso físico. De esta manera, se ha validado el sistema de diagnóstico a partir de una estructura similar a la que tendría una aplicación virtual, como por ejemplo de entrenamiento de personal inexperto.

Tras dicha validación, se ha seguido la configuración de la Figura 7.6. El robot y los sensores y los actuadores del sistema de manufactura han dejado de ser emulados y han intercambiado señales de E/S con el **PLC**. Asimismo, se ha agregado un **HMI** para la más pruebas de interacción con el personal. Se ha integrado la plataforma en el funcionamiento normal del proceso industrial para su diagnóstico contextual. Igual que en la fase o aplicación anterior, el acceso de la plataforma a los datos de proceso e interacción del **PLC** ha sido realizado a través de **OPC UA**. En estos datos se ha basado la identificación de tareas y procedimientos.

7.2.2.5. Cierre del experimento

Una vez integrado en el marco un **PLC** a modo de sistema interactivo, y siendo posible diagnosticar pasos y situaciones, se ha dado fin a la experimentación descrita en este capítulo con la realización de una serie de ensayos. De cara a testar la plataforma en aspectos como el orden de pasos en una secuencia, la duración de los pasos y la duración de la secuencia en su conjunto, se ha hecho operar a la célula robótica alternando los modos de trabajo automático y manual. Se han forzado errores de forma

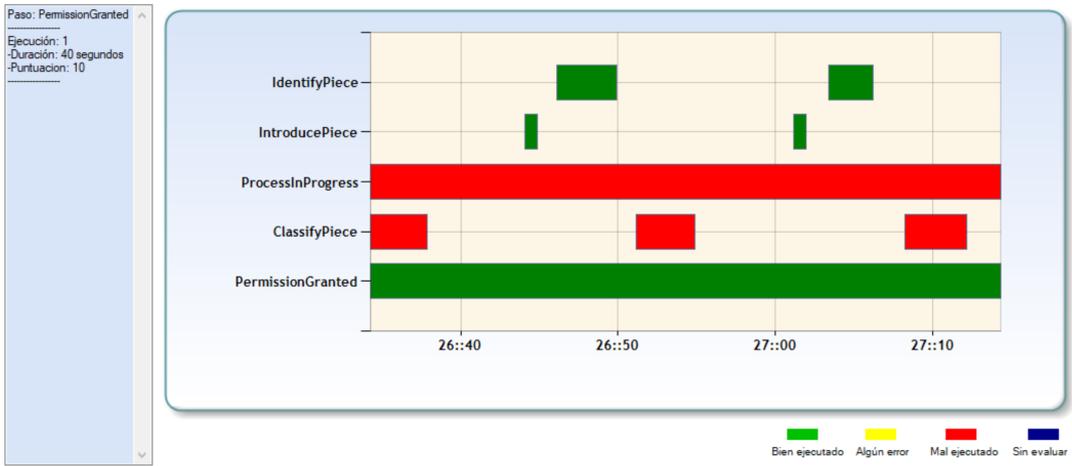


FIGURA 7.7: Secuencia incorrecta, y exceso de duración de un paso.

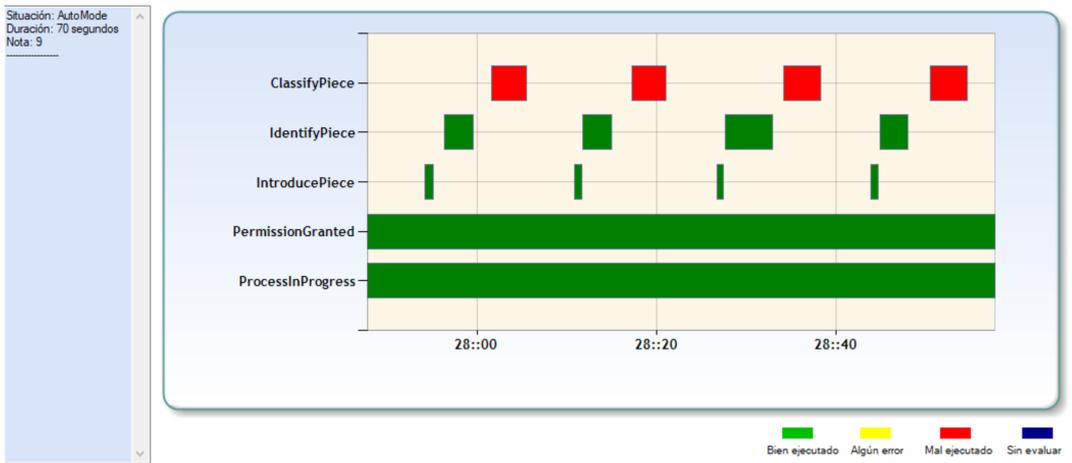


FIGURA 7.8: Exceso de duración de un paso.



FIGURA 7.9: Secuencia correcta, pero exceso de duración total.

periódica y aleatoria, mediante la inserción de pasos largos y secuencias incorrectas en cuanto al orden de los estos.

Se ha basado el diagnóstico en medir la duración de los pasos y comprobar el orden en que se dan. Se trata de dos variables definidas sin ningún esfuerzo de programación. La generación de observaciones ha sido programada automáticamente y a través del servidor [OPC UA](#).

7.2.3. Resultados

En lo que respecta a la operación en modo automático, la Tabla 7.2 resume qué se ha observado, interpretado y diagnosticado. La herramienta PATH ha hecho posible esto último tanto en tiempo de ejecución como en forma de informe de cierre de ensayo.

El diagnóstico al cierre del ensayo ha traído consigo la identificación de las situaciones reproducidas correctamente, así como la detección de la anomalía que se ha dado en las erróneas:

- La Figura 7.7 muestra que la secuencia de pasos de la operación en modo automático es incorrecta, a lo que añadir el exceso de la duración establecida para uno de dichos pasos, concretamente el de clasificación de la carcasa.
- En la Figura 7.8 se puede apreciar el exceso de duración del paso de clasificación de la carcasa.
- La Figura 7.9 corresponde a un exceso de duración total de la secuencia de operación en modo automático, siendo correcta la serie de pasos de los que se compone, así como la duración de los mismos.

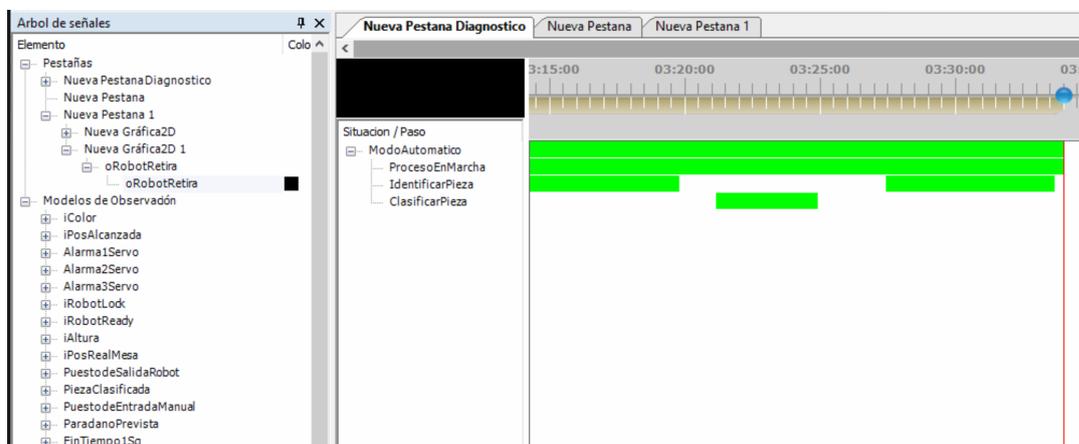


FIGURA 7.10: Interpretación en tiempo de ejecución.

Por otro lado, la Figura 7.10 muestra el diagnóstico en tiempo de ejecución de la situación de operación en modo automático y los pasos de los que consta.

7.2.4. Discusión

Este experimento ha tenido como objetivo la integración de un sistema interactivo, en este caso un [PLC](#), a través del estándar de comunicaciones [OPC UA](#), en un marco de diagnóstico de actividad. De los resultados obtenidos se deduce que, previa

TABLA 7.2: Caracterización de los subsistemas en el caso de uso: situación de operación en modo automático.

<p>Observación</p> <p>Eventos en el sistema interactivo (servidor OPC UA). Se combinarán en restricciones asociadas a pasos en situaciones.</p>	<p>Observaciones:</p> <ul style="list-style-type: none"> - Posición del selector. - Solicitud de acceso. - Orden de inicio. - Consigna de posición.
<p>Interpretación</p> <p>Reconocimiento del contexto a partir de las observaciones. Cumplimiento de restricciones para pasos y situaciones.</p>	<p>Situación: operación en modo automático.</p> <ul style="list-style-type: none"> - Paso: solicitud de permiso de acceso al recinto para situar carcasa. - Paso: dar inicio al proceso. - Paso: identificación de color y altura de la carcasa. - Paso: clasificación de la carcasa.
<p>Diagnóstico</p> <p>Corrección de situaciones y pasos.</p> <ul style="list-style-type: none"> - Orden de pasos en secuencia. - Duración de paso. - Duración de secuencia. 	<p>Diagnosticado (✓ / -):</p> <ul style="list-style-type: none"> - ✓ - ✓ - ✓

reproducción periódica y aleatoria de errores en el programa del dispositivo de control, se han identificado las diferentes tareas definidas, y se ha detectado cuándo estas no se han realizado según lo establecido.

Como debilidad de lo propuesto cabe reseñar dicha manipulación del código del PLC para la inserción de errores. Se ha recurrido a temporizaciones de espera y la alteración del orden de las etapas de una secuencia automática modelada a partir de la metodología GRAFCET. Esto puede considerarse una manera mejorable de provocar errores aleatorios y periódicos en el funcionamiento del sistema, ya que se altera el código del dispositivo de control.

Se concluyen las siguientes respuestas a las RQs enunciadas para este experimento:

RQ06

¿Puede integrarse un sistema interactivo con conectividad OPC UA en un marco genérico de diagnóstico contextual de actividad a partir de dotar a este último de dicha conectividad?

Si, dotar de conectividad al marco genérico de diagnóstico mediante un cliente OPC UA posibilita el intercambio de señales con el sistema interactivo o PLC a través de dicho estándar, comúnmente utilizado por la nueva generación de dispositivos de control.

RQ07

¿Constituye un marco genérico de diagnóstico contextual de actividad, al que se ha integrado como sistema interactivo un PLC que controla un sistema industrial automatizado emulado a través de su DT, una herramienta virtual para entrenamiento/formación de personal inexperto?

Si, el marco hace posible el diagnóstico de actividad de aplicaciones virtuales, o la puesta en servicio de aplicaciones reales, con un PLC como parte de una estructura para la VC de sistemas industriales automatizados a modo de sistema interactivo. Se observan, interpretan y diagnostican las señales intercambiadas por el dispositivo de control y el DT del sistema.

RQ08

¿Constituye un marco genérico de diagnóstico contextual de actividad, al que se ha integrado como sistema interactivo un PLC que controla un sistema industrial automatizado, una herramienta para diagnóstico de actividad en aplicaciones industriales reales?

Si, el marco hace posible el diagnóstico de actividad de aplicaciones reales, con un PLC a modo de sistema interactivo. Se observan, interpretan y diagnostican las señales intercambiadas por el dispositivo de control y el sistema industrial automatizado.

7.3. Trabajo relacionado

Se ha procedido a la divulgación de este trabajo y sus resultados en un artículo de conferencia [151] (ver sección 8.2.1). Asimismo, se está trabajando para la inclusión

de la interacción de la persona con el sistema.

7.4. Conclusión y trabajo futuro

Se ha propuesto un marco de trabajo, **ADAPT**, para (i) el diagnóstico contextual de la ejecución de tareas automáticas y manuales en sistemas industriales automatizados, (ii) la definición de procedimientos basados en la interacción de personal experimentado con el sistema de control, y (iii) la formación de noveles. La contribución de **ADAPT** radica en la integración, en un marco genérico que se había utilizado en otros ámbitos, de cualquier sistema interactivo a través del estándar de comunicaciones **OPC UA**, el más usual en el citado campo de aplicación.

Se ha diseñado un caso de uso en el que el sistema interactivo es un **PLC** que controla una célula robótica, tanto virtual como física. El dispositivo de control se comunica a través del protocolo en cuestión tanto con la plataforma de diagnóstico como con un **DT** que emula el sistema de manufactura. Se ha programado la generación de observaciones de forma automática y sin esfuerzo de programación a través del servidor **OPC UA**.

Los resultados obtenidos demuestran (i) la correcta integración de cualquier sistema interactivo con la citada conectividad y (ii) la validez del marco en lo que se refiere al diagnóstico de contextual de situaciones y pasos en sistemas industriales automatizados.

La aportación de la aplicación de tecnologías de digitalización consiste en la reproducción de situaciones y escenarios de interés sin afectar al ritmo productivo del sistema real ni comprometer la integridad de equipamiento y personas:

- La emulación puede formar parte de una aplicación exclusivamente virtual para la formación de personal inexperto. Sin el uso de un **DT**, dicho aprendizaje tendría lugar en el sistema físico, con la problemática descrita.
- Asimismo, la emulación puede apoyar y hacer seguro el desarrollo, la configuración y la validación de aplicaciones diagnóstico que posteriormente se ejecuten con el sistema físico.

El estudio de **ADAPT** se ha basado principalmente en datos de proceso procedentes de un **PLC**, aunque en futuros trabajos también se incluirán datos de interacción proceso-operador procedentes de diferentes dispositivos **HMI**. Esta información proporcionará una visión más completa de la corrección del cumplimiento de la tarea, y se podrán abordar aplicaciones como la formación del operario.

Parte III

Cierre de la tesis

Capítulo 8

Conclusiones

Este capítulo cierra la tesis *Aplicación de tecnologías de digitalización a sistemas industriales automatizados basados en controladores lógicos programables*. Se le da inicio con un resumen de su contribución, la cual incluye la revisión de hipótesis de trabajo y su validación, así como las limitaciones de las propuestas realizadas. Se sigue con la divulgación de dicha contribución, tanto en lo que respecta a las publicaciones realizadas como en lo que se refiere a actividades derivadas de la misma. Tras el repaso al conocimiento adquirido mediante este trabajo, se concluye este capítulo con la investigación futura a la que se prevé que dé pie esta tesis.

8.1. Contribución

Esta sección tiene por objeto revisar la aportación de este trabajo a la comunidad científica en general y a la actividad de [MU-GEP](#) y su entorno en particular, a partir de la aplicación de tecnologías de digitalización a sistemas controlados por [PLC](#).

En primera instancia, ante la problemática a la que se enfrentan los colectivos de alumnado y de profesorado en la impartición de contenidos sobre automatización industrial en el aula académica, y que es extensible al personal técnico de desarrollo de soluciones para industria, se ha propuesto la digitalización para mejorar la formación en los términos descritos. Se ha diseñado un experimento consistente en la implementación del sistema de control para un sistema de manufactura, en el que a parte del alumnado se le ha facilitado un [DT](#) del proceso, para testar tanto la validez de la herramienta como los resultados detectados en la revisión de la literatura. Con ello se ha constituido un marco de trabajo para la [VC](#) apoyado en este tipo de herramientas, que ha dado pie a seguir experimentando en el resto de campos descritos.

La segunda contribución parte de la experimentación con una nueva manera de implementar [DEDSs](#) en [PLCs](#) de modo sencillo y a partir del enfoque [PN](#), amplio y con potentes reglas de validación formal. Se ha presentado una metodología para desarrollar soluciones automatizadas con escasa carga de trabajo de desarrollo de código y mínimos errores potenciales. Hay adiciones o modificaciones que requieren únicamente la edición gráfica de la estructura de la red, su validación automática y transferencia de los nuevos datos al [PLC](#). Esto puede permitir mayor velocidad en la programación y en la puesta en marcha de instalaciones. Los resultados han indicado, además, que el algoritmo en el que se basa la metodología presentada no consume más recursos de CPU que otros enfoques similares, así como su sencilla portabilidad entre plataformas de distintos fabricantes. Nótese que para quienes estén habituados a trabajar con [GRAFNET](#), el cambio de paradigma no es traumático, ya que se necesitan habilidades de modelado similares.

La tercera y última aportación se basa en una metodología propuesta para el diagnóstico contextual de actividad en un sistema industrial. En ella, cualquier dispositivo o aplicación con conectividad **OPC UA** puede hacer de sistema interactivo integrado en un marco genérico del que se ha dispuesto con anterioridad. Se ha experimentado con un **PLC** que intercambia señales de E/S con (i) un sistema automatizado, observando, interpretando y diagnosticando lo que sucede en el mismo, y (ii) un **DT** de dicho sistema que posibilita tanto el estado de la aplicación anterior como formar parte de una exclusivamente virtual para formación de personal.

En consecuencia, y en términos generales, de esta tesis se concluye que las tecnologías de digitalización de sistemas industriales automatizados atenúan/resuelven los problemas identificados, los cuales afronta el personal técnico a lo largo de todo el ciclo de vida útil de los mismos, y se considera que suponen un avance en los aspectos y en los términos descritos.

8.1.1. Validación de hipótesis

En esta sección se analiza la posible validación de las hipótesis de investigación del Capítulo 3 a partir de las contribuciones realizadas.

La primera de ellas tiene por enunciado *"El uso de un **DT** de un sistema industrial automatizado basado en **PLC** reduce la duración de la fase de puesta en servicio de su código y permite obtener resultados de mayor calidad, con menor disponibilidad de equipamiento y menor riesgo para el mismo y para las personas"*. Se propone el uso de la emulación para validar software de controladores antes de la puesta en servicio, agilizándola. El caso de estudio ha consistido en un experimento a partir del cual se comparan desarrollos apoyados en un **DT** con implementaciones convencionales, obteniéndose mejores resultados en los primeros en términos de duración del proyecto y calidad de la implementación.

La segunda hipótesis es enunciada como *"La definición matemática formal de una **PN** y la amplia teoría de validación estructural de la misma hacen posible implementar **DEDSs** en **PLCs** utilizando bloques de programa mayoritariamente fijos, independientes de la aplicación y del dispositivo, y sin consumir más recursos de CPU que otros enfoques similares"*. Se plantea el enfoque **PN**, más general que **GRAFNET** y con una amplia teoría de validación formal, como base de una metodología para la implementación de **DEDSs** en **PLCs**. La estructura de la red, previamente validada, es gestionada por bloques de programa fijos, teniendo únicamente que programar y validar manualmente aquellos correspondientes a la interpretación de la misma, lo cual supone un avance respecto a **GRAFNET**, ampliamente utilizado en el ámbito industrial, y dada la ausencia de formas sistemáticas para aplicar **PNs**. El caso de uso ha consistido en la automatización de un proceso virtual ejemplo típico de aplicación del enfoque **PN**, en el que se han implementado la metodología propuesta y otros enfoques similares. De los resultados obtenidos se deduce que no se requieren más recursos de CPU, y que el código es sencillo de transferir entre dispositivos de distintas marcas.

Se enuncia la tercera hipótesis como *"El uso de un **DT** hace posible validar nuevos desarrollos contra sistemas no disponibles físicamente de una manera visual y ágil"*. Como se ha avanzado en la anterior, se ha propuesto testar la metodología mencionada contra un sistema de manufactura ampliamente presente en la literatura para aplicar **PNs**, y no disponible físicamente. La emulación ha hecho posible validar el trabajo

en un sistema específico de una manera más visual, ágil y con menor riesgo de dejar situaciones sin testar, en comparación con el convencional forzado de valores en memorias de entradas y observación del comportamiento de las salidas.

La cuarta hipótesis tiene por enunciado "*La integración, a modo de sistema interactivo y través de OPC UA en un IS, de un PLC que controla un sistema industrial automatizado emulado a través de su DT, hace posible el diagnóstico de actividad en aplicaciones virtuales en un contexto de entrenamiento/formación de personal inexperto*". Se ha planteado dotar de conectividad a través de dicho estándar a un marco genérico de diagnóstico, lo cual ha hecho posible la integración de un PLC en el mismo, a modo de sistema interactivo, y que intercambia señales de E/S con un sistema industrial emulado. El conjunto constituye una herramienta exclusivamente virtual, por ejemplo, para formar personas sin comprometer su seguridad, la del equipamiento, o la producción. El caso de uso ha estado basado en una célula de manufactura emulada en la que se han diagnosticado con éxito situaciones de varios tipos.

La quinta hipótesis es enunciada como "*La integración, a modo de sistema interactivo y través de OPC UA en un IS, de un PLC que controla un sistema industrial automatizado, hace posible el diagnóstico de actividad en aplicaciones industriales reales*". En la línea del anterior, pero con el PLC intercambiando señales de E/S con el sistema industrial real, en este caso la mencionada célula de manufactura. Se han diagnosticado con éxito, en ambos casos, secuencias de pasos tanto en orden como en duraciones parciales y totales.

Se enuncia la sexta hipótesis como "*La estimación de consumos energéticos en modelos virtuales de nivel componente posibilita el ajuste de parámetros operativos a la máxima eficiencia para los ratios productivos previstos en sistemas industriales automatizados dotados de control de velocidad*". Se propone la emulación nivel componente como herramienta de estimación de consumos. Integrada en un marco para la puesta en servicio virtual de sistemas de control, hace posible determinar las consignas de velocidad que, cumpliendo con los ratios productivos previstos, supongan el menor gasto posible, evitando cuellos de botella y almacenamiento intermedio de material. De esta manera se evita realizar pruebas en el equipamiento recién puesto en marcha. Se ha trabajado con un sistema de manufactura exclusivamente virtual, como primera aproximación, y se han obtenido los consumos estimados para distintas consignas de velocidad, a partir de lo cual se puede dar con las más adecuadas para cumplir de la manera más eficiente con los ritmos de trabajo requeridos.

8.1.2. Limitaciones de las propuestas

En esta sección se discuten las limitaciones de las soluciones propuestas al ponerlas en práctica. En términos generales, y no directamente ligada al trabajo realizado sino al uso de DTs en general, aflora la dificultad en poder cuantificar con precisión las mejoras relativas al uso de una herramienta de emulación. Se ha adoptado un enfoque cualitativo con objeto de obtener resultados concluyentes y representativos.

En cuanto a la metodología de implementación de PNs en PLCs, cuenta con una debilidad que destaca del resto. La programación de la interpretación de la red, única parte a trabajar en el entorno de desarrollo, se realiza a criterio del personal técnico. Es decir, no se presenta una forma sistemática de hacerlo, ni se dispone, de momento, de una herramienta amigable o visual para ello. Además, se identifican otras posibles mejoras como la parametrización del algoritmo para una portabilidad entre plataformas más sencilla, o la extensión de la propuesta a enfoques de PNs

de alto nivel como una donde se adjunta información a cada marca, aumentando la capacidad de descripción de la red.

En lo relativo a la herramienta de diagnóstico a partir de la integración de cualquier dispositivo con conectividad **OPC UA** en un **IS**, se han insertado errores aleatorios y periódicos en el funcionamiento a partir de la manipulación del código del **PLC**. Debería explorarse la posibilidad de testar situaciones anómalas a partir de manipular la emulación del proceso pero no el sistema de control del mismo.

En lo que respecta al uso de modelos nivel componente para estimar consumos energéticos, la aplicación realizada es didáctica y manual. Se trata de un trabajo todavía en desarrollo y automatizable, pero en el que se estima que un **DT** puede apoyar el uso de técnicas de optimización en la definición de parámetros operativos en sistemas industriales automatizados.

8.2. Divulgación de la contribución

En esta sección se listan las acciones de divulgación del trabajo realizado y sus resultados. Se abre la misma con las publicaciones tanto de conferencias internacionales como de revistas, ya referidas en los capítulos correspondientes. Posteriormente, se dan a conocer actividades como ponencias, colaboraciones y redacción de propuestas.

8.2.1. Publicaciones

[C1] Artículo de conferencia [117] relativo a la contribución presentada en el Capítulo 5.

Virtual commissioning of a robotic cell: an educational case study.

Igor Azkarate, Mikel Ayani (HiS) y Luka Eciolaza.

ETFFA 2019

24th IEEE Conference on Emerging Technologies and Factory Automation.

Zaragoza, 10-13 de septiembre de 2019.

.....

[C2] Capítulo de libro [118] correspondiente a la contribución descrita en el Capítulo 5.

Benefits of using a digital twin in industrial automation learning activities.

Igor Azkarate

CIDICO 2019

I Congreso Internacional de Innovación Docente e Investigación en Educación Superior.

Madrid, 20-22 de noviembre de 2019.

.....

[C3] Artículo de conferencia [135] asociado a la contribución detallada en el Capítulo 6.

Petri net implementation in programmable logic controllers: methodology for development and validation.

Igor Azkarate, Juan Carlos Mugarza y Luka Eciolaza.

SAMI 2021

IEEE 19th World Symposium on Applied Machine Intelligence and Informatics.

Herl'any (Eslovaquia), 21-23 de enero de 2021. Online.

.....
[J1] Artículo de revista [136] relativo a la contribución presentada en el Capítulo 6.

Petri net-based semi-compiled code generation for programmable logic controllers.

Igor Azkarate, Juan Carlos Mugarza, Mikel Ayani (HiS) y Luka Eciolaza.

Recent Advances in Petri Nets Modeling, número especial de Applied Sciences (ISSN 2076-3417), del *Multidisciplinary Digital Publishing Institute (MDPI)*.

3 de agosto de 2021

.....
[C4] Artículo de conferencia [151] correspondiente a la contribución descrita en el Capítulo 7.

ADAPT: an Automatic Diagnosis of Activity and Processes in auTomation environments.

Igor Azkarate, Aitor Aguirre, Josu Uranga y Luka Eciolaza.

ETFA 2020

25th IEEE Conference on Emerging Technologies and Factory Automation.

Viena (Austria), 8-11 de septiembre de 2020. Online.

8.2.2. Actividades

A continuación se detallan las ponencias realizadas y las colaboraciones puestas en marcha, así como las propuestas redactadas en el contexto de este trabajo de tesis.

8.2.2.1. Ponencias

Workshops sobre *S3D*, con la presencia de Mikel Ayani, *director ejecutivo o Chief Executive Officer (CEO)* de Simumatik AB:

- Jornada de 4 horas con alumnado de 2º de Grado de Ingeniería en Electrónica Industrial de *MU-GEP* (26 de marzo de 2019).
- Jornada de 4 horas con alumnado de 3º de Grado de Ingeniería en Electrónica Industrial de *MU-GEP* (29 de marzo de 2019).

8.2.2.2. Proyecto educativo de colaboración entre centros: Gemelo Digital: Didacticación de generación y uso de modelos virtuales de sistemas automatizados (2018/19)

Participantes:

- Se trata de una colaboración entre centros de **Formación Profesional (FP)**, en la que se ha trabajado conjuntamente con el **Centro Integrado de Formación Profesional (CIFFP) Tartanga**.

Trabajo:

- Familiarización con el software **S3D**.
- Generación de material didáctico sobre generalidades y la herramienta **S3D**.
- Desarrollo de gemelos digitales en **S3D**.

Transferencia:

- Talleres en el aula.
- Blog: <http://berrikuntza.blog.tartanga.eus/gemelo-digital/>
- Presentación en la jornada de transferencia de proyectos de centros y áreas de especialización de **FP** del 6 de junio de 2019, con el objetivo de compartir una visión general del trabajo que se está llevando a cabo en el campo de la innovación en el País Vasco, fomentando así las sinergias entre centros y reforzando el sistema de innovación de la **FP**.

8.2.2.3. Colaboración en el Área de Innovación Tecnológica y Sistemas Inteligentes de Tknika, Centro de Investigación Aplicada de FP Euskadi (2019/Act.)

Trabajo:

- Testado de distintas herramientas de emulación de sistemas industriales automatizados. Pruebas de digitalización de maquetas del laboratorio.
- Generación de material didáctico sobre generalidades y las herramientas Simumatik Platform y Factory I/O: configuración, conexión a controladores reales o emulados, y uso de gemelos digitales públicos.

Transferencia:

- Impartición del curso *INDUSTRIA 4.0: Gemelos digitales, puesta en marcha virtual de proyectos de automatización (Virtual Commissioning)* del 8 al 26 de marzo de 2021. Dirigido a profesorado de **FP**, se siguió el formato webinar para un total de 36 horas de dedicación por parte de las personas matriculadas.

8.2.2.4. Redacción de propuestas

Durante el desarrollo de esta tesis, el doctorando ha participado activamente en solicitudes de subvención de proyectos. Sirva como ejemplo DIGIPACK, presentado el 22 de febrero de 2018 junto con **UPTC** en la convocatoria ELKARTEK del Gobierno Vasco.

En esta ocasión, al igual que en otras de características similares, se realizaron tareas como las que siguen:

- Lectura y análisis de la convocatoria.
- Definición y desarrollo de los objetivos y enfoques necesarios para dar respuesta a los objetivos científico-tecnológicos y criterios de evaluación de la convocatoria.

- Planificación del proyecto.
- Apoyo a la escritura de la memoria.

8.3. Competencias desarrolladas y conocimiento adquirido

En esta sección se resume el aprendizaje extraído de esta tesis.

8.3.1. Capacitación preliminar

En la fase de capacitación se ha asistido a diferentes formaciones, que han tenido lugar en el Campus de Arrasate-Mondragón de [MU-GEP](#) y que han servido para adquirir los conocimientos necesarios para abordar un proceso de tesis doctoral.

Durante el primer trimestre del año académico 2017/18:

- Gestión de proyectos de investigación (3 ECTS).
- Pautas metodológicas para la realización de una tesis doctoral (3 ECTS).
- Producción de textos científicos I (1 ECTS).

Durante el segundo trimestre del año académico 2017/18:

- Innovación y emprendimiento (2 ECTS).
- Investigación en red (3 ECTS).
- Producción de textos científicos II (1 ECTS).
- Técnicas documentales de investigación (1 ECTS).

Además de las citadas formaciones del plan de doctorandos de [MU-GEP](#), se ha trabajado en la adquisición de la base necesaria en varios conceptos, algunos de ellos directamente relacionados con los experimentos presentados en este documento:

- AMESIN, MCD, Process Simulate y Plant Simulate de Siemens, en forma de una jornada a medida acerca de la [VC](#), en virtud de un convenio de colaboración firmado con [MU-GEP](#), el cual incluye formación de personal.
- B&R Automation Studio y conceptos de *motion control* que imparte personal técnico de B&R en el Máster en Automatización Industrial, en virtud de otro convenio de colaboración.
- [S3D](#) y Simumatik Platform, en forma de asesoramiento, al tener contacto con el equipo de Simumatik AB.
- Formación relativa al nuevo [laboratorio ciber-físico o Cyber-Physical LAB \(CP LAB\)](#) y las tecnologías asociadas al mismo.

8.3.2. Capacitación durante el desarrollo de la tesis

Antes de nada, y obviamente, la serie de lecciones aprendidas que se describen a continuación han ido acompañadas de la adquisición de la base técnica necesaria, tanto en lo que respecta a conceptos como en lo que se refiere al manejo de herramientas software, desarrollo de código de [PLC](#), etc.

En el período abarcado por esta tesis se han comprobado y aprendido numerosas cuestiones relacionadas con la aplicación de tecnologías de digitalización en sistemas industriales automatizados controlados por PLC, como por ejemplo, y como anticipaba la revisión de la literatura, que un proyecto apoyado por un DT hace posible la validación del sistema de control antes de su puesta en servicio, acortando plazos y obteniendo resultados de mayor calidad. La base adquirida y el material generado han resultado de utilidad para el resto de experimentos, así como para la actividad docente e investigadora del grupo de investigación.

Se han comparado, además, los conocimientos previos relativos al enfoque GRAFCET para la implementación de DEDSs en PLCs con las PNs, herramienta de modelado más general y con una amplia teoría de validación formal. Esta ha servido para proponer una metodología de aplicación de PNs sencilla, con escaso desarrollo de código y de carácter semi-compilado. En esta ocasión no se han vuelto a comprobar las virtudes de utilizar un DT para la puesta en servicio virtual previa a la real, sino para que se ha validado de una manera exclusivamente virtual de un nuevo desarrollo de I+T contra un sistema de manufactura concreto y no disponible. Es decir, una herramienta de digitalización permite testar desarrollos en maquinaria a la carta sin necesidad de invertir en la misma, reproduciendo todas las situaciones que se estimen necesarias.

Asimismo, se ha tenido ocasión de aprender que, contando con conectividad OPC UA tanto dispositivos de control como emuladores de procesos industriales, pueden integrarse estos en ISs y diseñarse un entorno de diagnóstico de actividad válido para su aplicación tanto en el sistema físico como para formar sin riesgo a personal sin experiencia.

Finalmente, se ha tomado nota de la aportación del modelado nivel componente, que hace posible tareas más allá de la validación de programas de control, como por ejemplo la estimación de consumos energéticos en receptores como motores asíncronos trifásicos. Esto permite al personal técnico ajustar las consignas de velocidad al ritmo productivo requerido, ahorrando energía y reduciendo costes derivados tanto de esta como de la existencia de cuellos de botella y la necesidad de espacio para almacenamiento intermedio.

8.4. Perspectivas y trabajo futuro

8.4.1. Transferencia a industria

La actividad investigadora de MU-GEP está orientada a industria. Se mantiene contacto fluido con empresas desarrolladoras de soluciones automatizadas y hay, por parte del grupo de investigación, una colaboración con UPTC y UHS, líderes en maquinaria profesional para envase y embalaje y en ingeniería logística dirigida al ámbito de la distribución y de la fabricación automática, respectivamente. Estas organizaciones han mostrado interés tanto en la aplicación de tecnologías de digitalización como en el marco PN aplicado a la implementación de DEDSs en PLCs. Se prevé seguir trabajando en estos ámbitos, solicitando subvenciones a proyectos, generando actividad investigadora y divulgando los resultados.

8.4.2. Aplicación de los métodos propuestos

Al hilo de lo anterior, y a medio plazo, hay interés por probar un posible cambio de paradigma de GRAFCET a PN en el desarrollo de DEDSs, testando la metodología

propuesta en maquinaria real de UHS, previa puesta en servicio virtual.

Se quiere, asimismo, dotar de mayor potencial al marco de diagnóstico de actividad presentado.

8.4.3. Líneas de investigación futuras

Identificado el potencial tanto de la metodología basada en PNs como del marco de diagnóstico de actividad, se contemplan las siguientes posibilidades en ambos ámbitos:

- Propuesta y lanzamiento de trabajos de tesis, para dar continuidad a lo realizado, si se cuenta con personas candidatas y la financiación necesaria.
- Como se ha mostrado en limitaciones de los métodos y en el estado del arte, se prevé la mejora o resolución de las mismas, según lo detallado en cada uno de los experimentos.
- Lo anterior incluye la adaptación de las metodologías a otras herramientas o entornos, así como el diseño de casos de estudio adecuados para la evaluación empírica.

Parte IV

Bibliografía

Bibliografía

- [1] *Grupos de Investigación y Transferencia - Mondragon Unibertsitatea*, 27-07-2022. dirección: <https://www.mondragon.edu/es/investigacion/ingenieria-tecnologia/grupos-investigacion-transferencia/-/mu-inv-mapping/grupo/robotica-y-automatizacion>.
- [2] M. Akerman, «Towards interoperable information and communication systems for manufacturing operations,» Tesis doct., jun. de 2016.
- [3] R. David, «Petri Nets and Grafcet for Specification of Logic Controllers,» *IFAC Proceedings Volumes*, vol. 26, págs. 683-688, jul. de 1993. DOI: [10.1016/S1474-6670\(17\)49215-9](https://doi.org/10.1016/S1474-6670(17)49215-9).
- [4] G. Michel y F. Duncan, «Programmable logic controllers: architecture and applications,» 1990.
- [5] L. Ben-naoum, R. Boel, L. Bongaerts et al., «Methodologies for discrete event dynamic systems: A survey,» vol. 36, jul. de 2000.
- [6] F. DiCesare, G. Harhalakis, J.-M. Proth, M Silva y F. Vernadat, *Practice of Petri nets in manufacturing*. Springer, 1993.
- [7] T. Murata, «Petri nets: Properties, analysis and applications,» *Proceedings of the IEEE*, vol. 77, n.º 4, págs. 541-580, 1989.
- [8] M. Silva y E. Teruel, «Petri nets for the design and operation of manufacturing systems,» *European journal of control*, vol. 3, n.º 3, págs. 182-199, 1997.
- [9] M. Silva y E. Teruel, «A Systems Theory Perspective of Discrete Event Dynamic Systems: The Petri Net Paradigm,» ago. de 1996.
- [10] M. V. Moreira y J. C. Basilio, «Bridging the Gap Between Design and Implementation of Discrete-Event Controllers,» *IEEE Transactions on Automation Science and Engineering*, vol. 11, n.º 1, págs. 48-65, 2014. DOI: [10.1109/TASE.2013.2281733](https://doi.org/10.1109/TASE.2013.2281733).
- [11] G. Cansever e I. B. Kucukdemiral, «A new approach to supervisor design with sequential control Petri-net using minimization technique for discrete event system,» *The International Journal of Advanced Manufacturing Technology*, vol. 29, n.º 11-12, págs. 1267-1277, 2005. DOI: [10.1007/s00170-005-0010-4](https://doi.org/10.1007/s00170-005-0010-4).
- [12] M. Stanton, W. Arnold y A. Buck, «Modelling and Control of Manufacturing Systems Using Petri Nets,» *IFAC Proceedings Volumes*, vol. 29, n.º 1, págs. 4754-4759, 1996. DOI: [10.1016/s1474-6670\(17\)58432-3](https://doi.org/10.1016/s1474-6670(17)58432-3).
- [13] H. Lasi, P. Fettke, H.-G. Kemper, T. Feld y M. Hoffmann, «Industry 4.0,» *Business & Information Systems Engineering*, vol. 6, n.º 4, págs. 239-242, 2014. DOI: [10.1007/s12599-014-0334-4](https://doi.org/10.1007/s12599-014-0334-4).
- [14] S. Phuyal, D. Bista y R. Bista, «Challenges, Opportunities and Future Directions of Smart Manufacturing: A State of Art Review,» *Sustainable Futures*, vol. 2, pág. 100 023, 2020. DOI: [10.1016/j.sftr.2020.100023](https://doi.org/10.1016/j.sftr.2020.100023).
- [15] M. Ghobakhloo, «Industry 4.0, digitization, and opportunities for sustainability,» *Journal of Cleaner Production*, vol. 252, pág. 119 869, 2020. DOI: [10.1016/j.jclepro.2019.119869](https://doi.org/10.1016/j.jclepro.2019.119869).

- [16] D. Guo, M. Li, Z. Lyu et al., «Synchroperation in industry 4.0 manufacturing,» *International Journal of Production Economics*, vol. 238, pág. 108 171, 2021. DOI: [10.1016/j.ijpe.2021.108171](https://doi.org/10.1016/j.ijpe.2021.108171).
- [17] W. Kritzinger, M. Karner, G. Traar, J. Henjes y W. Sihn, «Digital Twin in manufacturing: A categorical literature review and classification,» *IFAC-PapersOnLine*, vol. 51, n.º 11, págs. 1016-1022, 2018. DOI: [10.1016/j.ifacol.2018.08.474](https://doi.org/10.1016/j.ifacol.2018.08.474).
- [18] T. H.-J. Uhlemann, C. Schock, C. Lehmann, S. Freiburger y R. Steinhilper, «The Digital Twin: Demonstrating the potential of real time data acquisition in production systems,» vol. 9, págs. 113-120, 2017, ISSN: 2351-9789. DOI: [10.1016/j.promfg.2017.04.043](https://doi.org/10.1016/j.promfg.2017.04.043).
- [19] E. Negri, L. Fumagalli y M. Macchi, «A Review of the Roles of Digital Twin in CPS-based Production Systems,» *Procedia Manufacturing*, vol. 11, págs. 939-948, 2017. DOI: [10.1016/j.promfg.2017.07.198](https://doi.org/10.1016/j.promfg.2017.07.198).
- [20] J. Lee, B. Bagheri y H.-A. Kao, «A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems,» *Manufacturing Letters*, vol. 3, págs. 18-23, 2015. DOI: [10.1016/j.mfglet.2014.12.001](https://doi.org/10.1016/j.mfglet.2014.12.001).
- [21] S. Boschert y R. Rosen, «Digital twin—the simulation aspect,» en *Mechatronic Futures*, Springer, 2016, págs. 59-74.
- [22] G. N. Schroeder, C. Steinmetz, C. E. Pereira y D. B. Espindola, «Digital Twin Data Modeling with AutomationML and a Communication Methodology for Data Exchange,» vol. 49, págs. 12-17, 2016, ISSN: 2405-8963. DOI: [10.1016/j.ifacol.2016.11.115](https://doi.org/10.1016/j.ifacol.2016.11.115).
- [23] M. N. Asmat, S. U. R. Khan y S. Hussain, «Uncertainty handling in cyber-physical systems: State-of-the-art approaches, tools, causes, and future directions,» *Journal of Software: Evolution and Process*, 2022. DOI: [10.1002/smr.2428](https://doi.org/10.1002/smr.2428).
- [24] I. McGregor, «The relationship between simulation and emulation,» en *Proceedings of the Winter Simulation Conference*, vol. 2, 2002, 1683-1688 vol.2. DOI: [10.1109/WSC.2002.1166451](https://doi.org/10.1109/WSC.2002.1166451).
- [25] F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang y F. Sui, «Digital twin-driven product design, manufacturing and service with big data,» *The International Journal of Advanced Manufacturing Technology*, vol. 94, n.º 9-12, págs. 3563-3576, 2017. DOI: [10.1007/s00170-017-0233-1](https://doi.org/10.1007/s00170-017-0233-1).
- [26] J. Vachalek, L. Bartalsky, O. Rovny, D. Sismisova, M. Morhac y M. Loksik, «The digital twin of an industrial production line within the industry 4.0 concept,» 2017. DOI: [10.1109/pc.2017.7976223](https://doi.org/10.1109/pc.2017.7976223).
- [27] P. Hoffmann, T. M. Maksoud, R. Schumann y G. C. Premier, «Virtual Commissioning Of Manufacturing Systems A Review And New Approaches For Simplification,» en *ECMS 2010 Proceedings edited by A Bargiela S A Ali D Crowley E J H Kerckhoffs*, ECMS, 2010. DOI: [10.7148/2010-0175-0181](https://doi.org/10.7148/2010-0175-0181).
- [28] M. Dahl, K. Bengtsson, M. Fabian y P. Falkman, «Automatic Modeling and Simulation of Robot Program Behavior in Integrated Virtual Preparation and Commissioning,» *Procedia Manufacturing*, vol. 11, págs. 284 -291, 2017, 27th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017, 27-30 June 2017, Modena, Italy, ISSN: 2351-9789. DOI: [https://doi.org/10.1016/j.promfg.2017.07.107](https://doi.org/https://doi.org/10.1016/j.promfg.2017.07.107). dirección: <http://www.sciencedirect.com/science/article/pii/S2351978917303116>.
- [29] M. Oppelt y L. Urbas, «Integrated virtual commissioning an essential activity in the automation engineering process,» 2014. DOI: [10.1109/iecon.2014.7048867](https://doi.org/10.1109/iecon.2014.7048867).

- [30] M. Dahl, K. Bengtsson, P. Bergagård, M. Fabian y P. Falkman, «Integrated Virtual Preparation and Commissioning: supporting formal methods during automation systems development,» *IFAC-PapersOnLine*, vol. 49, n.º 12, págs. 1939 -1944, 2016, 8th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2016, ISSN: 2405-8963. DOI: <https://doi.org/10.1016/j.ifacol.2016.07.914>. dirección: <http://www.sciencedirect.com/science/article/pii/S2405896316312022>.
- [31] M. Johnstone, D. Creighton y S. Nahavandi, «Enabling industrial scale simulation / emulation models,» en *2007 Winter Simulation Conference*, IEEE, 2007. DOI: [10.1109/wsc.2007.4419701](https://doi.org/10.1109/wsc.2007.4419701).
- [32] M. Schamp, S. Hoedt, A. Claeys, E. Aghezzaf y J. Cottyn, «Impact of a virtual twin on commissioning time and quality,» *IFAC-PapersOnLine*, vol. 51, n.º 11, págs. 1047 -1052, 2018, 16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018, ISSN: 2405-8963. DOI: <https://doi.org/10.1016/j.ifacol.2018.08.469>. dirección: <http://www.sciencedirect.com/science/article/pii/S2405896318315970>.
- [33] *Simumatik3D - Home*, 10-08-2020. dirección: <https://simumatik3d.github.io/>.
- [34] K. Herbuś, P. Ociepka y A. Gwiazda, «Virtual Activating of a Robotized Production Cell with Use of the Mechatronics Concept Designer Module of the PLM Siemens NX System,» en *Advances in Intelligent Systems and Computing*, Springer International Publishing, 2018, págs. 417-425. DOI: [10.1007/978-3-319-97490-3_41](https://doi.org/10.1007/978-3-319-97490-3_41).
- [35] L. V. Guerrero, V. V. López y J. E. Mejía, «Virtual Commissioning with Process Simulation (Tecnomatix),» *Computer-Aided Design and Applications*, vol. 11, n.º sup1, S11-S19, 2014. DOI: [10.1080/16864360.2014.914400](https://doi.org/10.1080/16864360.2014.914400). eprint: <https://doi.org/10.1080/16864360.2014.914400>. dirección: <https://doi.org/10.1080/16864360.2014.914400>.
- [36] *Simumatik - Supporting Learning in PLC Programming & Robotics*, 28-12-2020. dirección: <https://simumatik.com/>.
- [37] *CIROS® Education – Applying virtual learning environments - CIROS® - Software & E-Learning - Learning Systems - Festo Didactic*, 24-01-2020. dirección: <https://www.festo-didactic.com/int-en/learning-systems/software-e-learning/ciros/ciros-education-applying-virtual-learning-environments.htm?fbid=aW50LmVuLjU1Ny4xNy4xOC4xMTEwLjgxODc>.
- [38] *Festo Didactic*, 24-01-2020. dirección: <https://www.festo-didactic.com>.
- [39] *Next-gen plc training - factory i/o*, 10-08-2020. dirección: <https://factoryio.com/>.
- [40] E. Irigoyen, E. Larzabal y R. Priego, «Low-cost platforms used in Control Education: An educational case study,» *IFAC Proceedings Volumes*, vol. 46, n.º 17, págs. 256-261, 2013. DOI: [10.3182/20130828-3-uk-2039.00058](https://doi.org/10.3182/20130828-3-uk-2039.00058).
- [41] D. Jones, C. Snider, A. Nassehi, J. Yon y B. Hicks, «Characterising the Digital Twin: A systematic literature review,» *CIRP Journal of Manufacturing Science and Technology*, vol. 29, págs. 36-52, 2020. DOI: [10.1016/j.cirpj.2020.02.002](https://doi.org/10.1016/j.cirpj.2020.02.002).
- [42] M. Grieves, «Digital twin: manufacturing excellence through virtual factory replication,» *White paper*, vol. 1, n.º 2014, págs. 1-7, 2014.
- [43] E. Glaessgen y D. Stargel, «The digital twin paradigm for future NASA and US Air Force vehicles,» en *53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference 20th AIAA/ASME/AHS adaptive structures conference 14th AIAA*, 2012, pág. 1818.
- [44] M. Shafto, M. Conroy, R. Doyle, K. C. GE, J LeMoigne y L Wang, «NASA Modeling, Simulation,» *Information Technology & Processing*, 2010. dirección:

- https://www.nasa.gov/pdf/501321main_TA11-MSITP-DRAFT-Nov2010-A1.pdf.
- [45] R. Rosen, G. von Wichert, G. Lo y K. D. Bettenhausen, «About The Importance of Autonomy and Digital Twins for the Future of Manufacturing,» vol. 48, págs. 567-572, 2015, ISSN: 2405-8963. DOI: [10.1016/j.ifacol.2015.06.141](https://doi.org/10.1016/j.ifacol.2015.06.141).
- [46] M. Garetti, P. Rosa y S. Terzi, «Life cycle simulation for the design of product-service systems,» *Computers in Industry*, vol. 63, n.º 4, págs. 361-369, 2012.
- [47] D. M. D'Addona, A. M. M. S. Ullah y D. Matarazzo, «Tool-wear prediction and pattern-recognition using artificial neural network and DNA-based computing,» *Journal of Intelligent Manufacturing*, vol. 28, n.º 6, págs. 1285-1301, 2015. DOI: [10.1007/s10845-015-1155-0](https://doi.org/10.1007/s10845-015-1155-0).
- [48] R. Ružarovský, R. Holubek y D. R. D. Sobrino, «Virtual commissioning of a robotic cell prior to its implementation into a real flexible production system,» vol. 26, págs. 93-101, 2018, ISSN: 1338-0532. DOI: [10.2478/rput-2018-0011](https://doi.org/10.2478/rput-2018-0011).
- [49] S. T. Mortensen y O. Madsen, «A Virtual Commissioning Learning Platform,» *Procedia Manufacturing*, vol. 23, págs. 93-98, 2018. DOI: [10.1016/j.promfg.2018.03.167](https://doi.org/10.1016/j.promfg.2018.03.167).
- [50] K. Tebani, B. Mihoubi, M Kadri, G. Mehdi y B. Bouzouia, «Hardware-In-the-Loop Simulation for Validating PLC Programs,» nov. de 2015.
- [51] J. L. Vazquez-Gonzalez, «Virtual lab for industrial automation course,» vol. 14, pág. 4, ISSN: 1861-2121. DOI: [10.3991/ijoe.v14i08.8758](https://doi.org/10.3991/ijoe.v14i08.8758).
- [52] R. G. Lins, B. Guerreiro, R. Schmitt, J. Sun, M. Corazzim y F. R. Silva, «A novel methodology for retrofitting CNC machines based on the context of industry 4.0,» 2017. DOI: [10.1109/syseng.2017.8088293](https://doi.org/10.1109/syseng.2017.8088293).
- [53] M. Ayani, M Ganebäck y A. H.C. Ng, «Digital Twin: Applying emulation for machine reconditioning,» *Procedia CIRP*, vol. 72, págs. 243-248, ene. de 2018. DOI: [10.1016/j.procir.2018.03.139](https://doi.org/10.1016/j.procir.2018.03.139).
- [54] C. G. Lee y S. C. Park, «Survey on the virtual commissioning of manufacturing systems,» vol. 1, págs. 213-222, 2014, ISSN: 2288-4300. DOI: [10.7315/jcde.2014.021](https://doi.org/10.7315/jcde.2014.021).
- [55] J. Lorente Zamora y G. Sorarrain Aguirrezabala, *Modelling and emulation of industrial electric, pneumatic and hydraulic system*, 2018. DOI: [10.1016/j.eprs.2018.10.030](https://doi.org/10.1016/j.eprs.2018.10.030).
- [56] O. Mathias, W. Gerrit, D. Oliver, L. Benjamin, S. Markus y U. Leon, «Automatic Model Generation for Virtual Commissioning based on Plant Engineering Data,» *IFAC Proceedings Volumes*, vol. 47, n.º 3, págs. 11635 -11 640, 2014, 19th IFAC World Congress, ISSN: 1474-6670. DOI: <https://doi.org/10.3182/20140824-6-ZA-1003.01512>. dirección: <http://www.sciencedirect.com/science/article/pii/S1474667016434671>.
- [57] G. Reinhart y G. Wünsch, «Economic application of virtual commissioning to mechatronic production systems,» *Production Engineering*, vol. 1, n.º 4, págs. 371-379, 2007, ISSN: 1863-7353. DOI: [10.1007/s11740-007-0066-0](https://doi.org/10.1007/s11740-007-0066-0). dirección: <https://doi.org/10.1007/s11740-007-0066-0>.
- [58] Z. Liu, N. Suchold y C. Diedrich, «Virtual Commissioning of Automated Systems,» en *Automation, InTech*, 2012. DOI: [10.5772/45730](https://doi.org/10.5772/45730).
- [59] N. Shahim y C. Møller, «Economic Justification of Virtual Commissioning in Automation Industry,» en *Proceedings of the 2016 Winter Simulation Conference*, ép. WSC '16, Arlington, Virginia: IEEE Press, 2016, págs. 2430-2441, ISBN: 978-1-5090-4484-9. DOI: [10.1109/wsc.2016.7822282](https://doi.org/10.1109/wsc.2016.7822282). dirección: <http://dl.acm.org/citation.cfm?id=3042094.3042396>.

- [60] J. Oscarsson, M. A. Jeusfeld y A. Jenefeldt, «Towards Virtual Confidence - Extended Product Lifecycle Management,» págs. 708-717, 2016, ISSN: 1868-4238. DOI: [10.1007/978-3-319-33111-9_64](https://doi.org/10.1007/978-3-319-33111-9_64).
- [61] F. Auinger, M. Vorderwinkler y G. Buchtela, «Interface driven domain-independent modeling architecture for “soft-commissioning” and “reality in the loop”,» en *Proceedings of the 31st conference on Winter simulation Simulation—a bridge to the future - WSC '99*, ACM Press, 1999. DOI: [10.1145/324138.324504](https://doi.org/10.1145/324138.324504).
- [62] H. Schludermann, T. Kirchmair y M. Vorderwinkler, «Soft commissioning: Hardware-in-the-loop-based Verification of Controller Software,» en *Proceedings of the 32Nd Conference on Winter Simulation*, ép. WSC '00, Orlando, Florida: Society for Computer Simulation International, 2000, págs. 893-899, ISBN: 0-7803-6582-8. dirección: <http://dl.acm.org/citation.cfm?id=510378.510505>.
- [63] S. Boschert, C. Heinrich y R. Rosen, «Next generation digital twin,» en *Proc. tmce*, Las Palmas de Gran Canaria, Spain, vol. 2018, 2018, págs. 7-11.
- [64] U. Dahmen y J. Roßmann, «Simulation-based Verification with Experimentable Digital Twins in Virtual Testbeds,» págs. 139-147, 2018. DOI: [10.1007/978-3-662-56714-2_16](https://doi.org/10.1007/978-3-662-56714-2_16).
- [65] Y. Liao, H. Lee y K. Ryu, «Digital Twin concept for smart injection molding,» vol. 324, pág. 012077, 2018, ISSN: 1757-8981. DOI: [10.1088/1757-899x/324/1/012077](https://doi.org/10.1088/1757-899x/324/1/012077).
- [66] J. Rossmann, O. Stern y R. Wischnewski, «Systematics with a corresponding software tool for the integrated virtual commissioning of production lines from planning over to simulation to operation,» págs. 707-716, ene. de 2007.
- [67] Meyer, «Integrated Virtual Commissioning-Improved scalability by connecting material flow simulations and virtual machines,» vol. 70, 2013.
- [68] Dahl, «Sequence Planner: Supporting Integrated Virtual Preparation and Commissioning,» vol. 50, págs. 5818-5823, 2017, ISSN: 2405-8963. DOI: [10.1016/j.ifacol.2017.08.536](https://doi.org/10.1016/j.ifacol.2017.08.536).
- [69] L. Damiani, M. Demartini, P. Giribone, M. Maggiani, R. Revetria y F. Tonelli, «Simulation and Digital Twin Based Design of a Production Line: A Case Study,» en *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 2, 2018.
- [70] A. Vergnano, G. Berselli y M. Pellicciari, «Interactive simulation-based-training tools for manufacturing systems operators: an industrial case study,» vol. 11, págs. 785-797, 2017, ISSN: 1955-2513. DOI: [10.1007/s12008-016-0367-7](https://doi.org/10.1007/s12008-016-0367-7).
- [71] I. Calvo, F. López, E. Zulueta y P. González-Nalda, «Towards a methodology to build virtual reality manufacturing systems based on free open software technologies,» *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 11, n.º 3, págs. 569-580, 2016. DOI: [10.1007/s12008-016-0311-x](https://doi.org/10.1007/s12008-016-0311-x).
- [72] J. Pagan, «STEM Education Using Emulation Software for Hydraulic Fluid Power Applications,» vol. 6, págs. 86-92, 2018, ISSN: 2333-4495. DOI: [10.12691/jbms-6-3-4](https://doi.org/10.12691/jbms-6-3-4).
- [73] D. Velasquez, E. Perez, X. Oregui et al., «A Hybrid Machine-Learning Ensemble for Anomaly Detection in Real-Time Industry 4.0 Systems,» *IEEE Access*, vol. 10, págs. 72 024-72 036, 2022. DOI: [10.1109/access.2022.3188102](https://doi.org/10.1109/access.2022.3188102).
- [74] M. Vathoopan, M. Johny, A. Zoitl y A. Knoll, «Modular Fault Ascription and Corrective Maintenance Using a Digital Twin,» vol. 51, págs. 1041-1046, 2018, ISSN: 2405-8963. DOI: [10.1016/j.ifacol.2018.08.470](https://doi.org/10.1016/j.ifacol.2018.08.470).
- [75] D. Velásquez, A. Sánchez, S. Sarmiento et al., «A Cyber-Physical Data Collection System Integrating Remote Sensing and Wireless Sensor Networks for

- Coffee Leaf Rust Diagnosis,» *Sensors*, vol. 21, n.º 16, pág. 5474, 2021. DOI: [10.3390/s21165474](https://doi.org/10.3390/s21165474).
- [76] F. Tao, M. Zhang, Y. Liu y A. Y. C. Nee, «Digital twin driven prognostics and health management for complex equipment,» vol. 67, págs. 169-172, 2018, ISSN: 0007-8506. DOI: [10.1016/j.cirp.2018.04.055](https://doi.org/10.1016/j.cirp.2018.04.055).
- [77] W. Luo, T. Hu, C. Zhang e Y. Wei, «Digital twin for CNC machine tool: modeling and using strategy,» *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, n.º 3, págs. 1129-1140, 2018. DOI: [10.1007/s12652-018-0946-5](https://doi.org/10.1007/s12652-018-0946-5).
- [78] O. Freeman Gebler, B. Hicks, A. Harrison y M. Barker, «On the feasibility of inferring the applied mechanical loading of a conveyor system test rig from monitored system parameters,» English, COMADEM 2017 : 30th International Conference on Condition Monitoring and Diagnostic Engineering Management ; Conference date: 10-07-2017, jul. de 2017. dirección: <http://www.comadem2017.com/>.
- [79] Knapp, «Building blocks for a digital twin of additive manufacturing,» vol. 135, págs. 390-399, 2017, ISSN: 1359-6454. DOI: [10.1016/j.actamat.2017.06.039](https://doi.org/10.1016/j.actamat.2017.06.039).
- [80] E. J. Tuegel, A. R. Ingraffea, T. G. Eason y S. M. Spottswood, «Reengineering Aircraft Structural Life Prediction Using a Digital Twin,» *International Journal of Aerospace Engineering*, vol. 2011, págs. 1-14, 2011. DOI: [10.1155/2011/154798](https://doi.org/10.1155/2011/154798).
- [81] T. Javied, T. Rackow, R. Stankalla, C. Sterk y J. Franke, «A Study on Electric Energy Consumption of Manufacturing Companies in the German Industry with the Focus on Electric Drives,» *Procedia CIRP*, vol. 41, págs. 318 -322, 2016, Research and Innovation in Manufacturing: Key Enabling Technologies for the Factories of the Future - Proceedings of the 48th CIRP Conference on Manufacturing Systems, ISSN: 2212-8271. dirección: <http://www.sciencedirect.com/science/article/pii/S2212827115010586>.
- [82] I. Expósito e I. Mujika, *Reductions in Energy Consumption through Process Optimisation and Variable Production*, 2017. dirección: <http://urn.kb.se/resolve?urn=urn:nbn:se:his:diva-15209>.
- [83] H. Zipper, F. Auris, A. Strahilov y M. Paul, «Keeping the digital twin up-to-date — Process monitoring to identify changes in a plant,» en *2018 IEEE International Conference on Industrial Technology (ICIT)*, IEEE, 2018. DOI: [10.1109/icit.2018.8352419](https://doi.org/10.1109/icit.2018.8352419).
- [84] C. S. Longo y C. Fantuzzi, «Simulation and optimisation of production lines in the framework of the IMPROVE project,» en *2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, IEEE, 2017. DOI: [10.1109/etfa.2017.8247582](https://doi.org/10.1109/etfa.2017.8247582).
- [85] I. Ricondo, A. Porto y M. Ugarte, «A digital twin framework for the simulation and optimization of production systems,» *Procedia CIRP*, vol. 104, págs. 762-767, 2021. DOI: [10.1016/j.procir.2021.11.128](https://doi.org/10.1016/j.procir.2021.11.128).
- [86] M. C. Magnanini, O. Melnychuk, A. Yemane et al., «A Digital Twin-based approach for multi-objective optimization of short-term production planning,» *IFAC-PapersOnLine*, vol. 54, n.º 1, págs. 140-145, 2021. DOI: [10.1016/j.ifacol.2021.08.077](https://doi.org/10.1016/j.ifacol.2021.08.077).
- [87] R. David y H. L. Alla, *Du Grafset aux réseaux de Petri*. 1992.
- [88] S. G. Tzafestas, M. G. Pantelelis y D. L. Kostis, «Design and Implementation of a Logic Controller using Petri Nets and Ladder Logic Diagrams,» *Systems Analysis Modelling Simulation*, vol. 42, n.º 1, págs. 135-167, 2002. DOI: [10.1080/02329290212164](https://doi.org/10.1080/02329290212164).

- [89] S. Peng y M. Zhou, «Ladder Diagram and Petri-Net-Based Discrete-Event Control Design Methods,» *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, vol. 34, n.º 4, págs. 523-531, 2004. DOI: [10.1109/tsmcc.2004.829286](https://doi.org/10.1109/tsmcc.2004.829286).
- [90] J. Zaytoon y B. Riera, «Synthesis and implementation of logic controllers - A review,» *Annual Reviews in Control*, vol. 43, mar. de 2017. DOI: [10.1016/j.arcontrol.2017.03.004](https://doi.org/10.1016/j.arcontrol.2017.03.004).
- [91] M. Uzam y A. Jones, «Discrete event control system design using automation Petri nets and their ladder diagram implementation,» *International Journal of Advanced Manufacturing Technology*, vol. 14, págs. 716-728, ene. de 1998. DOI: [10.1007/BF01438224](https://doi.org/10.1007/BF01438224).
- [92] S. Peng y M. Zhou, «Sensor-based stage Petri net modelling of PLC logic programs for discrete-event control design,» *International Journal of Production Research - INT J PROD RES*, vol. 41, págs. 629-644, feb. de 2003. DOI: [10.1080/0020754021000042364](https://doi.org/10.1080/0020754021000042364).
- [93] M. U. Borges y E. J. Lima, «Conversion methodologies from Signal Interpreted Petri Nets to Ladder Diagram and C language in Arduino,» *International Journal of Mechanical Engineering Education*, vol. 46, n.º 4, págs. 302-314, 2018. DOI: [10.1177/0306419018759921](https://doi.org/10.1177/0306419018759921).
- [94] G. B. Lee, H. Zandong y J. S. Lee, «Automatic generation of ladder diagram with control Petri Net,» *Journal of Intelligent Manufacturing*, vol. 15, n.º 2, págs. 245-252, 2004. DOI: [10.1023/b:jims.0000018036.84607.37](https://doi.org/10.1023/b:jims.0000018036.84607.37).
- [95] G. Frey, «Automatic implementation of Petri net based control algorithms on PLC,» vol. 4, feb. de 2000, 2819 -2823 vol.4, ISBN: 0-7803-5519-9. DOI: [10.1109/ACC.2000.878725](https://doi.org/10.1109/ACC.2000.878725).
- [96] M. Heiner y T. Menzel, «A Petri Net Semantics for the PLC Language Instruction List *),» ene. de 1998.
- [97] S. Korotkin, G. Zaidner, B. Cohen, A. Ellenbogen, M. Arad e Y. Cohen, «A Petri Net formal design methodology for discrete-event control of industrial automated systems,» en *2010 IEEE 26-th Convention of Electrical and Electronics Engineers in Israel*, 2010, págs. 000 431-000 435.
- [98] M. Markiewicz y L. Gniewek, «A Program Model of Fuzzy Interpreted Petri Net to Control Discrete Event Systems,» *Applied Sciences*, vol. 7, n.º 4, 2017, ISSN: 2076-3417. DOI: [10.3390/app7040422](https://doi.org/10.3390/app7040422). dirección: <https://www.mdpi.com/2076-3417/7/4/422>.
- [99] S. Rezig, C. Ghorbel, Z. Achour y N. Rezg, «PLC-based implementation of supervisory control for flexible manufacturing systems using theory of regions,» *International Journal of Automation and Control*, vol. 13, n.º 5, págs. 619, 2019. DOI: [10.1504/ijaac.2019.101911](https://doi.org/10.1504/ijaac.2019.101911).
- [100] S. Makris, G. Michalos y G. Chryssolouris, «Virtual Commissioning of an Assembly Cell with Cooperating Robots,» *Advances in Decision Sciences*, vol. 2012, págs. 1-11, 2012. DOI: [10.1155/2012/428060](https://doi.org/10.1155/2012/428060).
- [101] R. Fleisch, D. Entner, T. Prante y R. Pfefferkorn, «Interactive Optimization of Path Planning for a Robot Enabled by Virtual Commissioning,» en *Computational Methods in Applied Sciences*, Springer International Publishing, 2018, págs. 339-354. DOI: [10.1007/978-3-319-89890-2_22](https://doi.org/10.1007/978-3-319-89890-2_22).
- [102] T. Stock y G. Seliger, «Opportunities of Sustainable Manufacturing in Industry 4.0,» *Procedia CIRP*, vol. 40, págs. 536 -541, 2016, 13th Global Conference on Sustainable Manufacturing – Decoupling Growth from Resource Use, ISSN: 2212-8271. DOI: <https://doi.org/10.1016/j.procir.2016>.

- 01.129. dirección: <http://www.sciencedirect.com/science/article/pii/S221282711600144X>.
- [103] Y. Koren, U. Heisel, F. Jovane et al., «Reconfigurable Manufacturing Systems,» *CIRP Annals*, vol. 48, n.º 2, págs. 527-540, 1999. DOI: [10.1016/S0007-8506\(07\)63232-6](https://doi.org/10.1016/S0007-8506(07)63232-6).
- [104] S. Konstantinov, M. Ahmad, K. Ananthanarayan y R. Harrison, «The Cyber-Physical e-machine Manufacturing System: Virtual Engineering for Complete Lifecycle Support,» vol. 63, págs. 119-124, 2017, ISSN: 2212-8271. DOI: [10.1016/j.procir.2017.02.035](https://doi.org/10.1016/j.procir.2017.02.035).
- [105] S. T. Mortensen, D. Chrysostomou y O. Madsen, «A novel framework for virtual recommissioning in reconfigurable manufacturing systems,» 2017. DOI: [10.1109/etfa.2017.8247744](https://doi.org/10.1109/etfa.2017.8247744).
- [106] A. Hengstebeck, A. Barthelmey y J. Deuse, «Reconfiguration Assistance for Cyber-Physical Production Systems,» págs. 177-186, 2018. DOI: [10.1007/978-3-662-56714-2_20](https://doi.org/10.1007/978-3-662-56714-2_20).
- [107] Z Aspar, N. Shaikh-Husin y M Khalil-Hani, «Algorithm to Convert Signal Interpreted Petri Net models to Programmable Logic Controller Ladder Logic Diagram Models,» *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 10, n.º 3, págs. 905-916, 2018.
- [108] A. Conde, K. L. de Ipiña, M. Larrañaga et al., «LAGUNTXO: A Rule-Based Intelligent Tutoring System Oriented to People with Intellectual Disabilities,» en *Visioning and Engineering the Knowledge Society. A Web Science Perspective*, Springer Berlin Heidelberg, 2009, págs. 186-195. DOI: [10.1007/978-3-642-04754-1_20](https://doi.org/10.1007/978-3-642-04754-1_20).
- [109] J. García, K. Lopez-de Ipiña, E. Irigoyen et al., «INTELLIGENT TUTORING SYSTEM TO INTEGRATE PEOPLE WITH DOWN SYNDROME INTO WORK ENVIRONMENTS,» jul. de 2006.
- [110] A. Alvarez-Alvarez, J. M. Alonso y G. Trivino, «Human activity recognition in indoor environments by means of fusing information extracted from intensity of WiFi signal and accelerations,» *Information Sciences*, vol. 233, págs. 162-182, 2013. DOI: [10.1016/j.ins.2013.01.029](https://doi.org/10.1016/j.ins.2013.01.029).
- [111] X. Gil, «Industria 4.0 apuesta por la sostenibilidad,» 2017, Fecha de consulta: noviembre de 2018. dirección: <https://peopleplusprofit.org/industria-4-0-apuesta-por-la-sostenibilidad/>.
- [112] S. Weyer, T. Meyer, M. Ohmer, D. Gorecky y D. Zühlke, «Future Modeling and Simulation of CPS-based Factories: an Example from the Automotive Industry,» *IFAC-PapersOnLine*, vol. 49, n.º 31, págs. 97 -102, 2016, 12th IFAC Workshop on Intelligent Manufacturing Systems IMS 2016, ISSN: 2405-8963. DOI: <https://doi.org/10.1016/j.ifacol.2016.12.168>. dirección: <http://www.sciencedirect.com/science/article/pii/S2405896316328397>.
- [113] P. Puntel-Schmidt y A. Fay, «Levels of Detail and Appropriate Model Types for Virtual Commissioning in Manufacturing Engineering,» *IFAC-PapersOnLine*, vol. 48, n.º 1, págs. 922 -927, 2015, 8th Vienna International Conference on Mathematical Modelling, ISSN: 2405-8963. DOI: <https://doi.org/10.1016/j.ifacol.2015.05.027>. dirección: <http://www.sciencedirect.com/science/article/pii/S2405896315000282>.
- [114] M. Martínez-Rojas, P.-F. María del Carmen, S.-H. Jose Manuel, J. M. Alonso Moral et al., «Desarrollo de un repositorio común de contenidos para facilitar el proceso enseñanza-aprendizaje,» 2018.

- [115] I. Calvo Gordillo, E. Zulueta Guerrero, U. Gangoiti Gurtubay y J. M. López Guede, «Laboratorios remotos y virtuales en enseñanzas técnicas y científicas,» 2008. dirección: http://www.ehu.eus/ikastorratza/3_alea/laboratorios.pdf.
- [116] S. Wilkinson, «Focus group methodology: a review,» *International Journal of Social Research Methodology*, vol. 1, n.º 3, págs. 181-203, 1998. DOI: [10.1080/13645579.1998.10846874](https://doi.org/10.1080/13645579.1998.10846874).
- [117] I. Azkarate Fernández, M. Ayani Eguía y L. Eciolaza Echeverría, «Virtual commissioning of a robotic cell: an educational case study,» en *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, IEEE, 2019, págs. 820-825.
- [118] VV.AA., *Innovación docente e investigación en Ciencias, Ingeniería y Arquitectura*, Dykinson, ed. 2019, ISBN: 978-84-1324-559-1.
- [119] T. Sobh, «Discrete Event Dynamic Systems: An Overview,» *Technical Reports (CIS)*, mayo de 1991.
- [120] C. A. Petri, «Kommunikation mit automaten,» 1962.
- [121] M. M. zu Hörste y E. Schnieder, «Modelling and simulation of train control systems using Petri nets,» en *FMrail workshop*, vol. 3, 1999.
- [122] D. Caihong, «Application of Petri net to fault diagnosis in satellite,» *Journal of systems engineering and electronics*, vol. 12, n.º 2, págs. 92-96, 2001.
- [123] G. Cavone, M. Dotoli y C. Seatzu, «A survey on Petri net models for freight logistics and transportation systems,» *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, n.º 6, págs. 1795-1813, 2017.
- [124] M. Silva y L. Recalde, «On fluidification of Petri Nets: from discrete to hybrid and continuous models,» *Annual Reviews in Control*, vol. 28, n.º 2, págs. 253-266, 2004.
- [125] H. J. Genrich y K. Lautenbach, «System modelling with high-level Petri nets,» *Theoretical computer science*, vol. 13, n.º 1, págs. 109-135, 1981.
- [126] K. Jensen, *Coloured Petri nets: basic concepts, analysis methods and practical use*. Springer Science & Business Media, 1997, vol. 1.
- [127] M. A. Marsan, «Stochastic Petri nets: an elementary introduction,» en *European workshop on applications and theory in Petri nets*, Springer, 1988, págs. 1-29.
- [128] J. L. Peterson, *Petri net theory and the modeling of systems*. Prentice Hall PTR, 1981.
- [129] W. Reisig, *Petri Nets: An Introduction*. Berlin, Heidelberg: Springer-Verlag, 1985, ISBN: 0387137238.
- [130] M. Silva, *Las Redes de Petri: en la Automática y la Informática*. Editorial Ac, 1985.
- [131] *Platform Independent Petri net Editor 2*, 24-01-2020. dirección: <http://pipe2.sourceforge.net/>.
- [132] N. J. Dingle, W. J. Knottenbelt y T. Suto, «PIPE2: A Tool for the Performance Evaluation of Generalised Stochastic Petri Nets,» *SIGMETRICS Perform. Eval. Rev.*, vol. 36, n.º 4, 34-39, mar. de 2009, ISSN: 0163-5999. DOI: [10.1145/1530873.1530881](https://doi.org/10.1145/1530873.1530881). dirección: <https://doi.org/10.1145/1530873.1530881>.
- [133] J. Petter y S. Sawicki, «A COMPARISON OF PETRI NET SIMULATION TOOLS,» 2017.
- [134] X. Ye, J. Zhou y X. Song, «On reachability graphs of Petri nets,» *Computers & Electrical Engineering*, vol. 29, n.º 2, págs. 263-272, 2003, ISSN: 0045-7906. DOI: [https://doi.org/10.1016/S0045-7906\(01\)00034-9](https://doi.org/10.1016/S0045-7906(01)00034-9). dirección: <https://www.sciencedirect.com/science/article/pii/S0045790601000349>.
- [135] I. Azkarate Fernández, J. C. Mugarza Cortabarría y L. Eciolaza Echeverría, «Petri net implementation in programmable logic controllers: methodology

- for development and validation,» en *2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMi)*, 2021, págs. 000 015-000 020. DOI: [10.1109/SAMI50585.2021.9378673](https://doi.org/10.1109/SAMI50585.2021.9378673).
- [136] I. Azkarate, M. Ayani, J. C. Mugarza y L. Eciolaza, «Petri Net-Based Semi-Compiled Code Generation for Programmable Logic Controllers,» *Applied Sciences*, vol. 11, n.º 15, 2021, ISSN: 2076-3417. DOI: [10.3390/app11157161](https://doi.org/10.3390/app11157161). dirección: <https://www.mdpi.com/2076-3417/11/15/7161>.
- [137] *03 Processing Grafcet | Computer Engineering | Computer Programming*, 24-01-2020. dirección: <https://es.scribd.com/document/140248558/03-Processing-Grafcet>.
- [138] J. Wan, S. Tang, Q. Hua, D. Li, C. Liu y J. Lloret, «Context-Aware Cloud Robotics for Material Handling in Cognitive Industrial Internet of Things,» *IEEE Internet of Things Journal*, vol. 5, n.º 4, págs. 2272-2281, 2018.
- [139] A. N. Lee y J. L. Martínez Lastra, «Enhancement of industrial monitoring systems by utilizing context awareness,» en *2013 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA)*, 2013, págs. 277-284.
- [140] A. Kuutti, A. Dvoryanchikova, A. Lobov, J. L. M. Lastra y T. Vantera, «A device configuration management tool for context-aware system,» en *IEEE 10th International Conference on Industrial Informatics*, 2012, págs. 10-15.
- [141] A. Aguirre, «Sistemas Interactivos Inteligentes de Ayuda al Aprendizaje de Destrezas Físicas,» Tesis doct., dic. de 2013.
- [142] A. Aguirre, A. Lozano-Rodero, L. M. Matey, M. Villamañe y B. Ferrero, «A Novel Approach to Diagnosing Motor Skills,» *IEEE Transactions on Learning Technologies*, vol. 7, n.º 4, págs. 304-318, 2014, ISSN: 2372-0050. DOI: [10.1109/TLT.2014.2340878](https://doi.org/10.1109/TLT.2014.2340878).
- [143] S. S. A. Naser, «ITSB: An Intelligent Tutoring System Authoring Tool,» *Journal of Scientific and Engineering Research*, vol. 3, n.º 5, 2016.
- [144] A. Aguirre, A. Lozano-Rodero, M. Villamañe, B. Ferrero y L. M. Matey, «OLYMPUS: An Intelligent Interactive Learning Platform for Procedural Tasks,» en *GRAPP/IVAPP*, 2012, págs. 543-550.
- [145] S. Ohlsson, «Constraint-based student modelling,» *Journal of Interactive Learning Research*, vol. 3, n.º 4, pág. 429, 1992.
- [146] B Ferrero, «DETECTive: un entorno genérico e integrable para diagnóstico de actividades de aprendizaje,» *Fundamento, diseño y evaluación [Tesis Doctoral]: Univ. of the Basque Country UPV/EHU*, 2004.
- [147] D. SUN, *Synchronization and Control of Multiagent Systems*, English, 1.ª ed. CRC Press, Taylor & Francis Group, nov. de 2010, ISBN: 978-1-4398-2047-6.
- [148] P. Leitão, P. Vrba y T. Strasser, «Multi-agent systems as automation platform for intelligent energy systems,» nov. de 2013, págs. 66-71. DOI: [10.1109/IECON.2013.6699113](https://doi.org/10.1109/IECON.2013.6699113).
- [149] F. Bellifemine, A. Poggi y G. Rimassa, «Developing multi-agent systems with a FIPA-compliant agent framework,» *Software: Practice and Experience*, vol. 31, n.º 2, págs. 103-128, 2001.
- [150] J. F. Allen, «Maintaining knowledge about temporal intervals,» *Communications of the ACM*, vol. 26, n.º 11, págs. 832-843, 1983.
- [151] I. Azkarate Fernández, A. Aguirre Ortuzar, J. Uranga Andrés y L. Eciolaza Echeverría, «ADAPT: an Automatic Diagnosis of Activity and Processes in auTomation environments,» en *2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, vol. 1, 2020, págs. 945-951. DOI: [10.1109/ETFA46521.2020.9211996](https://doi.org/10.1109/ETFA46521.2020.9211996).

- [152] M. Faccio y M. Gamberi, «Energy saving in operations management through variable-speed drive technology: environmental versus economic convenience,» *International Journal of Services and Operations Management*, vol. 26, n.º 1, págs. 68-96, 2017.
- [153] M. Soleimani, M. N. Alizadeh y M. Moallem, «Economical replacement decision for induction motors in industry,» en *2018 IEEE Texas Power and Energy Conference (TPEC)*, IEEE, 2018, págs. 1-6.
- [154] P. Aivaliotis, K. Georgoulas y G. Chryssolouris, «The use of Digital Twin for predictive maintenance in manufacturing,» *International Journal of Computer Integrated Manufacturing*, vol. 32, n.º 11, págs. 1067-1080, 2019.
- [155] D. Hauf, D. Meike, Paryanto y J. Franke, «Energy consumption modeling within the virtual commissioning tool chain,» en *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, 2015, págs. 1357-1362. DOI: [10.1109/CoASE.2015.7294287](https://doi.org/10.1109/CoASE.2015.7294287).
- [156] N. Karanjkar, A. Joglekar, S. Mohanty, V. Prabhu, D. Raghunath y R. Sundaresan, «Digital Twin for Energy Optimization in an SMT-PCB Assembly Line,» en *2018 IEEE International Conference on Internet of Things and Intelligence System (IOTAIS)*, 2018, págs. 85-89. DOI: [10.1109/IOTAIS.2018.8600830](https://doi.org/10.1109/IOTAIS.2018.8600830).

Parte V

Anexos

Anexo A

Optimización energética en el marco de puesta en servicio virtual de sistemas industriales automatizados.

En países industrializados, las emisiones de CO₂ debidas a la generación de la energía demandada por la industria suponen un 70 % del total [152]. Los actuadores hidráulicos, neumáticos y eléctricos trabajan a su velocidad nominal en sistemas industriales automatizados, aunque en estos puedan producirse cambios en los ritmos de producción. Los motores asíncronos trifásicos son el principal consumidor de energía, pudiendo suponer en 30 % del total en el sector [153], como se muestra en la Figura A.1. Estos motores, además de operar a su velocidad nominal, acostumbran a estar sobredimensionados.

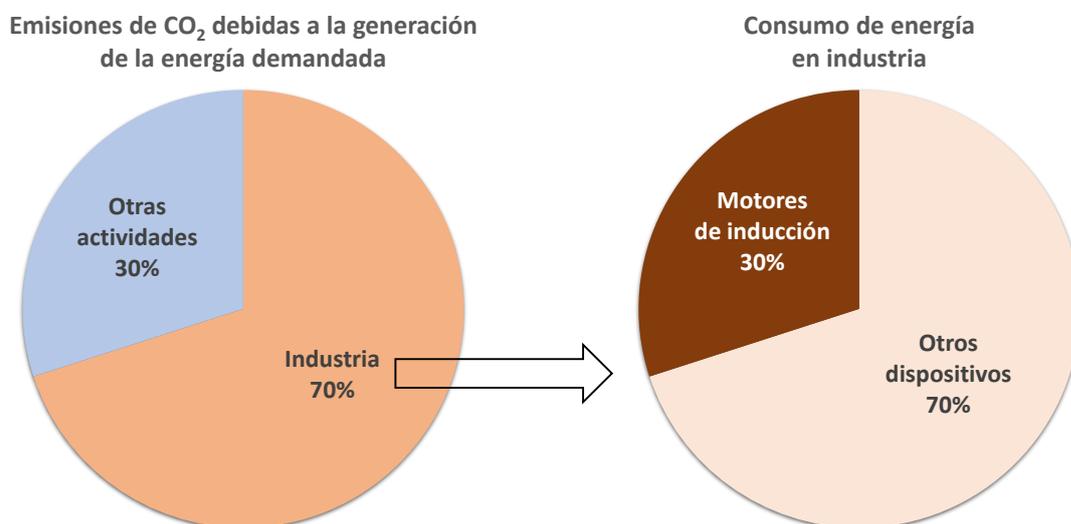


FIGURA A.1: Consumo energético debido a motores de inducción respecto al total de la demanda en países industrializados.

Datos extraídos de [152] y [153].

En comparación con el de nivel E/S, el mínimo requerido para el testado de software de control, el modelado nivel componente cuenta con potencial para apoyar otras tareas o fases. Se modela individualmente cada componente a partir de aspectos geométricos, cinemáticos, funcionales y de interfaz, pudiendo ser utilizado sin necesidad de conocimientos adicionales a los de automatización para apoyar, por

ejemplo, el diseño o la optimización energética. La estimación del consumo de un sistema industrial automatizado a través de su DT puede dar soporte al ajuste de parámetros del equipamiento antes de su envío a cliente.

Se anexa este capítulo a la memoria de la tesis por tratarse de un trabajo incipiente y que ha sido ideado durante el desarrollo de la misma. En el mismo se explora el beneficio de utilizar técnicas de optimización en la definición de parámetros operativos en sistemas industriales automatizados, y cómo un DT puede apoyar dicho ajuste, considerando tiempos de ciclo y los ratios de producción requeridos, y a partir del consumo de los receptores de un sistema emulado. Este dimensionado puede ser realizado durante el desarrollo de la solución automatizada, su VC, o en la puesta en servicio del equipamiento físico, sin necesidad de tests empíricos en la fase operativa del sistema.

A.1. Introducción

El punto de partida de esta sección es el sobredimensionado de motores y la selección de consignas de velocidad en VSDs, para pasar a valorar las posibilidades que ofrecen los DTs para dicho ajuste, y desembocar en cómo se puede dar respuesta a la problemática descrita.

A.1.1. El sobredimensionado de motores y la selección de consignas de variadores de velocidad

El sobredimensionado de motores y su funcionamiento a velocidad nominal supone un despilfarro energético y económico, así como la existencia de cuellos de botella en sistemas de manufactura, la necesidad de almacenamiento intermedio para material en proceso, y paradas en el ritmo productivo.

Dispositivos de tipo VSD surgen como alternativa al convencional automatismo basado en contactores que conmutan la alimentación de motores, y como elemento clave para el ahorro energético [152]. Los VSDs permiten ajustar la tensión y la frecuencia a aplicar a dicho receptor, y realizan los cambios al ritmo establecido por los parámetros de aceleración y desaceleración. Lo habitual es hacer trabajar a los motores por sistema bien a pleno rendimiento, bien a un valor cercano. Un ajuste más fino de consignas para un funcionamiento más eficiente en términos de consumo energético depende de pruebas empíricas tras la puesta en servicio del sistema de control. Es decir, tras dicha fase de puesta en marcha que puede traer consigo plazos de entrega más largos, con los costes asociados y el consiguiente impacto negativo en la imagen y en el posicionamiento de la marca desarrolladora, se pasa a una fase de optimización para, entre otros aspectos, la reducción de tiempo de ciclo de funcionamiento mediante pruebas que pueden afectar a la disponibilidad de máquina y a su producción.

A.1.2. Las posibilidades que ofrece la emulación

La irrupción de herramientas software para la emulación de sistemas industriales automatizados, así como la creciente capacidad de procesamiento de equipamiento informático, han traído consigo un cambio de paradigma en lo que se refiere a aplicaciones basadas en PLC. Los DTs dan soporte al desarrollo, testado y validación de código del dispositivo de control. Pero la emulación nivel componente no sólo da soporte a dichas pruebas relativas al código [154]. Hace posible aplicaciones de

optimización continua de procesos a partir de la modificación de parámetros y la observación del comportamiento sin ningún riesgo [26]. La eficiencia energética es considerada por [155] como criterio integrado en la fase de VC, donde utiliza modelos energéticos de robots industriales accionados por PLC. A partir de sensores instalados en la línea de producción física, [156] combina una plataforma de recogida de datos con la simulación basada en un DT del sistema automatizado, siendo ambas aplicaciones de código libre. Los modelos de consumo energético de [53] son exportados e integrados en el modelo virtual, el cual es ejecutado en un marco para la optimización multi-objetivo considerando consumo y ratio de producción.

A.1.3. Definición del problema

La mencionada existencia de motores trabajando sistemáticamente a su velocidad nominal, o de VSDs con las consignas sin ajustar, trae consigo cuellos de botella, necesidad de almacenaje intermedio y un exceso de gasto energético y, en consecuencia, económico. El ajuste empírico de dichas consignas tras la puesta en servicio del sistema de control de una solución automatizada requiere disponibilidad del equipamiento, retrasar la entrega al cliente, y una posible generación de defectivo.

En este capítulo se experimenta con la emulación nivel componente como herramienta para modelar el gasto energético en receptores de un sistema industrial automatizado. Las estimaciones tienen lugar en el modelo virtual, en el seno de un marco de VC de código de PLC, y no externamente. El DT puede dar soporte a la aplicación de técnicas de optimización para la determinación de parámetros operativos energéticamente óptimos para los requerimientos productivos, contribuyendo a un reto de presente y de futuro como es el descenso progresivo en el consumo.

A.2. Evaluación empírica

Para la evaluación de este experimento, que constituye una primera aproximación a la determinación de los parámetros operativos óptimos en términos energéticos sin afectar al ritmo productivo, se han enunciado las preguntas de investigación, para posteriormente caracterizar el ensayo a realizar. Finalmente, se han obtenido y discutido los resultados preliminares.

A.2.1. Preguntas de investigación

Las RQs a las que se ha pretendido dar respuesta son las que siguen:

RQ09

¿Es posible dotar a un DT de la estimación de consumos de receptores en modelos nivel componente?

RQ10

¿Puede un DT dar soporte al ajuste de parámetros operativos energéticamente óptimos a partir de la escritura de estos en el PLC a través de OPC UA, y la estimación de consumos de receptores en modelos nivel componente, para los ratios productivos deseados?

A.2.2. Caracterización del ensayo

En esta sección se detallan aspectos relativos al ensayo llevado a cabo, y está compuesta de cinco bloques. En el primero de ellos se enumera el material utilizado. Posteriormente se detallan las características del sistema emulado que se ha utilizado en el experimento. Se sigue detallando el trabajo realizado y en qué condiciones, así como la metodología seguida. La obtención de resultados supone el cierre.

A.2.2.1. Material utilizado

A continuación se listan las herramientas software requeridas para un desarrollo de estas características, especificándose lo que se ha utilizado para este caso en concreto. Nótese que, al tratarse de un primer estudio en este ámbito, el trabajo realizado es exclusivamente virtual, es decir, sin proceso físico ni su puesta en servicio. Se ha emulado un sistema industrial a través de su **DT** y se ha utilizado un **PLC** también emulado en su propio entorno de desarrollo. Se prevé la definición de un caso de uso más completo para otras fases de este trabajo.

- Herramienta de desarrollo y emulación de **PLCs** que cuenten con conectividad **OPC UA**.

Concretamente TIA Portal V15.1 de Siemens, con una CPU 1512C-1 PN emulada en PLCSIM Advanced V2.0 SP1. Se trata de herramientas con fuerte presencia en el mercado y para las cuales se cuenta con licencia en **MU-GEP**. Se dispone, además, de varias unidades del dispositivo, por sus características y por dicha conectividad, que serán utilizadas en otras fases de este trabajo.

- Herramienta para la emulación de sistemas industriales automatizados, con modelos nivel componente.

Se ha utilizado Simumatik Platform, software de apoyo a la **VC** de sistemas automatizados, así como al aprendizaje y entrenamiento de ingeniería eléctrica-electrónica y mecánica, programación de **PLCs** y robots, y otros conceptos técnicos. Basado en la emulación nivel componente, posibilita la estimación de consumos energéticos de receptores, entre otras funcionalidades.

- Cliente **OPC UA** comercial, para apoyar el desarrollo descrito a partir del acceso a variables en memoria del **PLC**.

Se ha recurrido a UaExpert 1.4.4.275, cliente **OPC UA** libre y con todas las prestaciones necesarias, para la lectura de datos del dispositivo.

- Un entorno integrado de desarrollo y aprendizaje o *Integrated Development and Learning Environment (IDLE)* de un lenguaje de programación de alto nivel, legible, reutilizable y sencillo de mantener.

Se ha optado por Python 3.8.6 y su **IDLE** 3.8.6 (64 bits), abierto, popular y fácil de utilizar, para el desarrollo y ejecución de aplicaciones que modifican parámetros del sistema de control y leen el consumo energético estimado para cada ensayo emulado en un **DT**, todo ello a través de **OPC UA**.

A.2.2.2. Caracterización del modelo virtual

En esta sección se introduce al lector a las características con las que cuenta el **DT** con el que se llevó a cabo este experimento, y que es el mostrado en la Figura A.2.

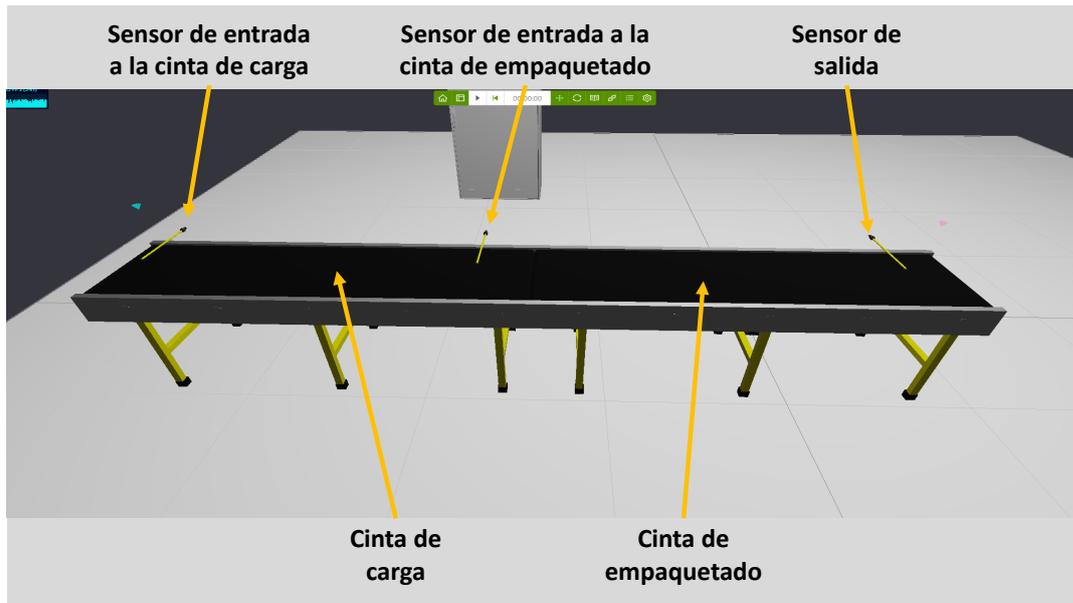


FIGURA A.2: Gemelo digital del sistema a automatizar.

A.2.2.2.1 Sistema objeto de estudio

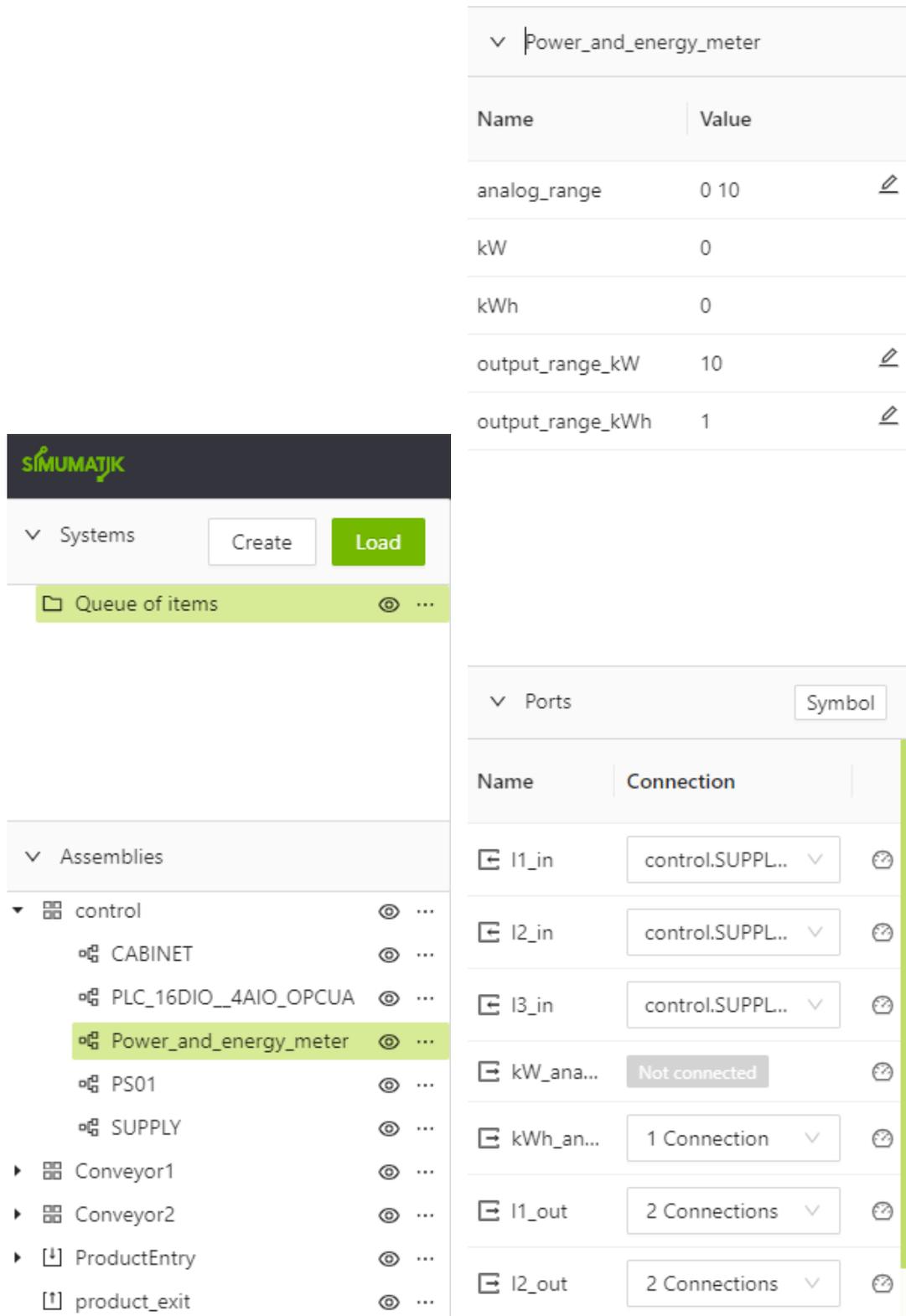
El sistema, exclusivamente emulado, utilizado en este experimento consta de dos cintas de transporte. Se ha escogido este sistema para las primeras pruebas por su sencillez, ya que hay únicamente dos parámetros del PLC a determinar (las consignas de velocidad) y tres entradas digitales a las que se hacen llegar las señales procedentes de los sensores. Además de compartir nada más que esas cinco variables del dispositivo de control con el DT, el código del mismo es simple, lo que permite centrar la atención en el ajuste de las consignas para accionar las cintas transportadoras de manera que se cumpla con el ratio productivo requerido con el menor gasto energético posible.

A.2.2.2.2 Diseño del modelo virtual

Se ha recurrido a Simumatik Platform, evolución de S3D, de la cual se cuenta con licencia en MU-GEP. El diseño del DT ha corrido a cargo del doctorando durante el experimento descrito en este capítulo, y previamente a las pruebas requeridas.

A.2.2.2.3 Particularidades del modelo virtual

El DT del sistema a parametrizar cuenta con el cableado virtual del sistema, incluyendo el de sensores/(pre)actuadores con entradas/salidas del objeto PLC. Se introduce material periódicamente a través del objeto gráfico situado en el extremo izquierdo, y sale agrupado en lotes de cuatro unidades por el lado opuesto. Para ello se recorren dos cintas transportadoras, de carga y empaquetado, respectivamente. Estas son accionadas por sendos servomotores de consigna de velocidad configurable. Hay, asimismo, tres sensores para la detección de presencia de material entrante, saliente y en una posición intermedia para la gestión de su agrupación en lotes.



(A) En la estructura de dispositivos.

(B) Detalle de su configuración.

FIGURA A.3: Contador de energía en el sistema emulado.

Estimación del consumo total

Se coloca, en la acometida de la instalación en el DT, un objeto para contaje de energía (Figura A.3). Insertado en la estructura de componentes (Figura A.3a), su configuración está basada en estas variables (Figura A.3b):

- Parámetros: (i) el rango de valores analógicos; (ii) la estimación de potencia y energía; y (iii) el rango en las respectivas salidas de potencia y energía.
- Puertos de entrada: las tres fases de la alimentación, desde el punto de acometida de la instalación.
- Puertos de salida: (i) las tres fases de la alimentación, al resto de la instalación; y (ii) las respectivas salidas analógicas relativas al consumo de potencia y energía.

La estimación de consumo se realiza a partir de la corriente calculada para cada receptor de la instalación.

Estimación de corriente en receptores

En esta aplicación hay, además de una fuente de alimentación, dos servomotores trifásicos (Figura A.5) que pueden ser controlados en velocidad o posición, a los que se les aplican consignas variables de velocidad. La Figura A.5a muestra uno de ellos insertado en la estructura de componentes. Las variables que intervienen en la configuración de un VSD son las de la Figura A.5b:

- Parámetros: (i) el rango de tensión utilizado para el control del dispositivo; (ii) velocidad mínima y máxima; (iii) la posición mínima y máxima; y (iv) la aceleración máxima.
- Puertos de entrada: (i) las tres fases de la alimentación; (ii) velocidad a alcanzar, en tensión; (iii) posición a alcanzar, en tensión; y (iv) modo de trabajo, posición o velocidad, configurable en tensión.
- Puertos de salida: (i) valor actual de posición o velocidad, según modo seleccionado; y (ii) indicador de que la consigna ha sido alcanzada.

La Figura A.4 muestra un detalle de cómo se opera con estas variables del VSD para, por ejemplo, estimar la corriente en receptores, en el editor de componentes de Simumatik Platform.

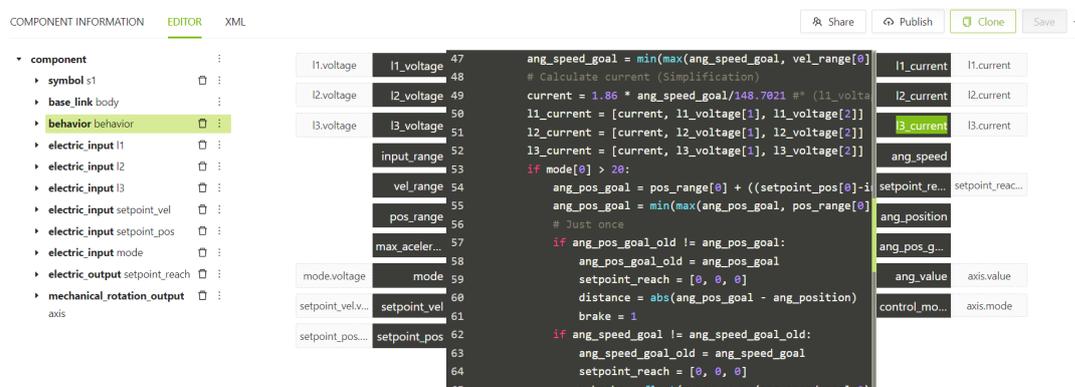
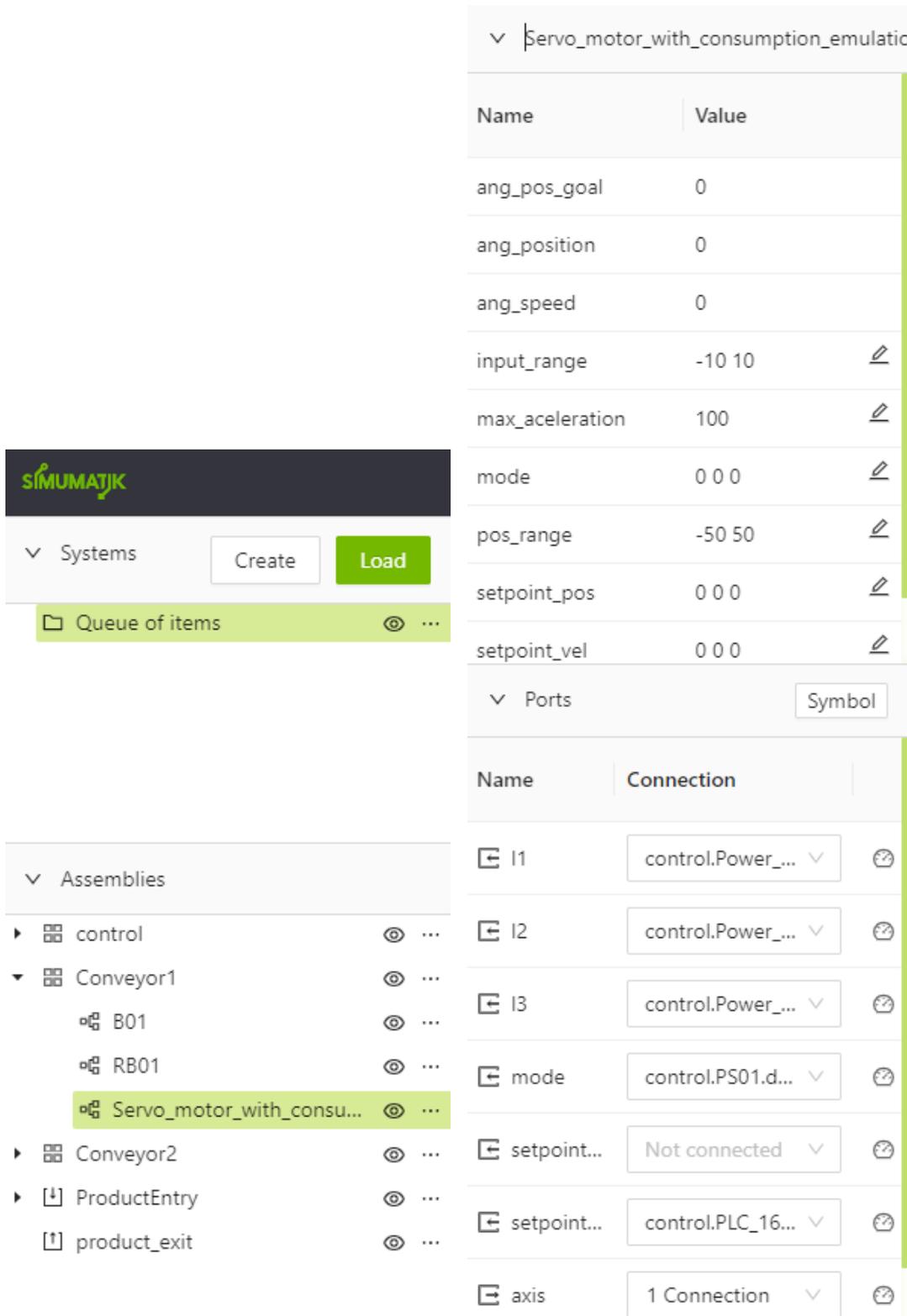


FIGURA A.4: Detalle del editor de componentes: comportamiento del servomotor.



(A) En la estructura de dispositivos.

(B) Detalle de su configuración.

FIGURA A.5: Servomotor en el sistema emulado.

A.2.2.2.4 Conectividad

Para la conexión de señales del dispositivo de control con el DT a través de OPC UA se han tenido en cuenta varios aspectos. En el caso del PLC, los siguientes:

- Se ha habilitado el servidor OPC UA de la CPU en una dirección IP y un puerto.
- Se ha asignado a todas las variables compartidas el mismo nombre que en el emulador.
- Se ha desactivado la actualización automática de memorias asociadas a entradas, para su escritura por parte del modelo virtual.

En lo que respecta al DT:

- En el objeto PLC del DT se hace referencia a la citada dirección IP y puerto del servidor OPC UA.

A.2.2.2.5 Configuración

Se ha dispuesto el sistema de control según una configuración SIL exclusivamente (ver Figura A.6).

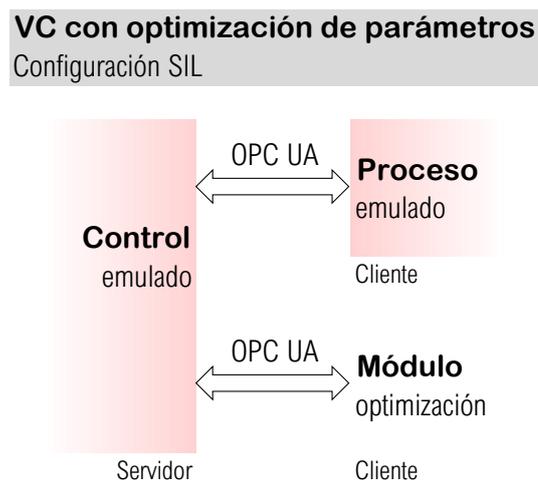


FIGURA A.6: Configuración del sistema de control.

A.2.2.3. Trabajo realizado

El trabajo preliminar realizado ha consistido en la integración de un módulo para la realización de ensayos y medidas de consumo energético en un marco para VC siguiendo la configuración mencionada.

La preparación del PLC ha constado de dos partes. La primera de ellas ha sido el desarrollo del código necesario para dar respuesta a las especificaciones de funcionamiento del sistema a automatizar. En la segunda se han incorporado, al código del PLC, variables de apoyo al experimento.

El programa de control realiza lo siguiente:

- Contaje asociado a cada uno de los tres sensores, cuya puesta a cero se realiza automáticamente cuando un lote completo haya salido del proceso.

- Movimiento de la primera cinta, la de carga, mientras el número de unidades detectadas en el sensor intermedio sea inferior a las detectadas en el de entrada.
- Movimiento de la segunda cinta, la de empaquetado:
 - Mientras se detecte pieza en el sensor de entrada y al menos una haya atravesado el sensor intermedio, para originar un hueco.
 - Cuando un lote completo haya alcanzado el sensor intermedio pero no el de salida, para la evacuación de todo el lote.

Las variables de apoyo al experimento añadidas al proyecto del PLC han sido estas:

- Indicador de ciclo de trabajo en curso, para la detección de sus flancos de activación y desactivación.

El PLC escribe su valor (verdadero o falso) a partir de las señales de E/S compartidas con en DT, y el módulo de optimización u *Optimization Module (OM)* lo lee.

- Consumo del último ciclo de trabajo completado.
El DT escribe su valor, y el OM lo lee.
- Duración del último ciclo completado o valor instantáneo de la duración del ciclo el curso.

El PLC la calcula a partir de los cambios en el indicador, y el OM lo lee.

El trabajo realizado en el DT ha consistido en la incorporación de un objeto de medida de consumo en la acometida de la alimentación eléctrica del modelo virtual. Se calcula la energía consumida a partir de la corriente total, la cual es estimada a partir de la combinación de las corrientes estimadas en cada uno de los subsistemas receptores. Se conecta la salida de este dispositivo de medida a una entrada analógica del PLC virtual, de igual manera que se hace tanto con sensores y entradas digitales, como con salidas digitales y preactuadores.

Posteriormente, se ha puesto en servicio el marco para VC (Figura A.7) en configuración SIL, ejecutando ciclos consecutivos en el DT (cliente OPC UA) comandado por el PLC (servidor OPC UA) emulado. A dicho marco se ha incorporado el OM (cliente OPC UA). Este módulo escribe consignas de velocidad para los dos VSDs y lee las variables del PLC ya enumeradas, de manera que se registran dichas consignas, el consumo estimado y la duración del ciclo para cada test. Ese registro hace posible la determinación de los parámetros idóneos para dar respuesta a los ratios productivos requeridos de la manera más eficiente. En caso de disponer del sistema físico, se utilizarían dichos parámetros en la puesta en servicio del sistema.

A.2.2.4. Metodología

A continuación se describe la metodología seguida en este capítulo y aplicada al experimento en cuestión. Se describen los pasos llevados a cabo para determinar, con el soporte de un DT del sistema a automatizar, los parámetros operativos para un funcionamiento energéticamente óptimo.

Una herramienta de emulación nivel componente en la que se estimen intensidades de corriente y consumos en receptores hace posible el desarrollo de este trabajo de

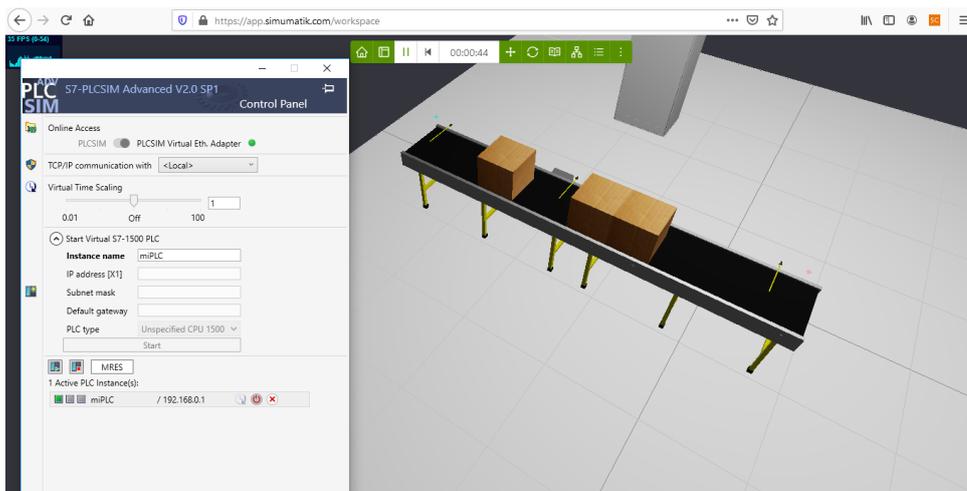


FIGURA A.7: Puesta en servicio virtual del sistema automatizado.

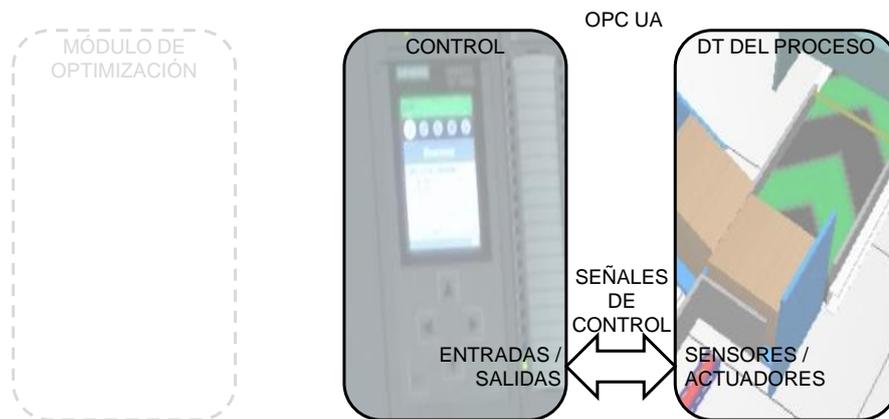


FIGURA A.8: Configuración base para puesta en servicio virtual.

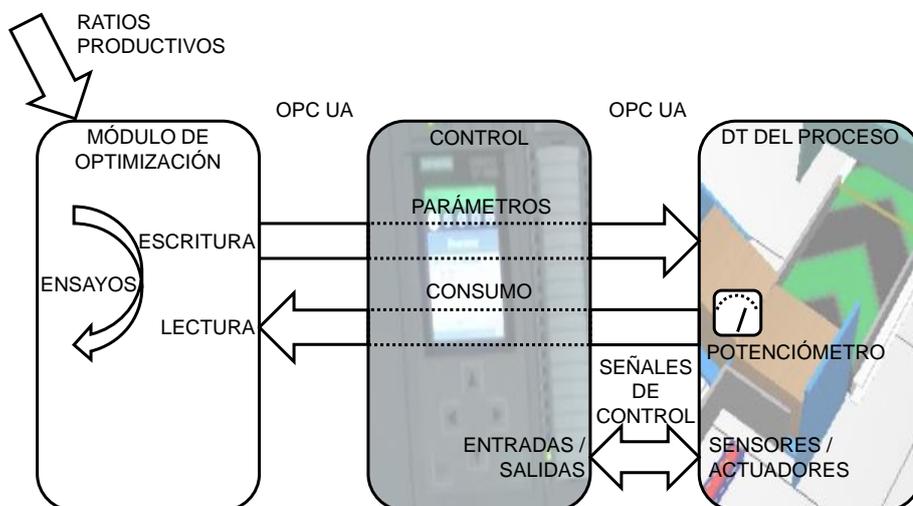


FIGURA A.9: Integración de un módulo de optimización.

una manera sencilla, sin esfuerzo de modelado y realizando lecturas directas mediante un objeto de tipo potenciómetro.

A.2.2.4.1 Configuración: marco para VC con integración de OM

El punto de partida es un marco convencional para VC (Figura A.8), en el que se conecta un DT del proceso al PLC, siendo este último físico o emulado en su entorno de desarrollo. El dispositivo de control envía señales de comando y recibe señales acerca del estado del sistema a través del estándar OPC UA. Es el proceso emulado, y no el ciclo de scan del controlador, el que escribe valores en memoria de entradas de este, a partir de lo que determinen los sensores virtuales.

Se incorpora a dicha configuración base un OM (Figura A.9) con objeto de minimizar el consumo energético del sistema industrial automatizado sin afectar a su ritmo productivo, a partir del protocolo de comunicación mencionado. El módulo en cuestión, configurado como cliente OPC UA y basado en un lenguaje de alto nivel, legible, reutilizable y sencillo de mantener, permite realizar ensayos a partir de la aplicación de combinaciones de parámetros y la lectura tanto del consumo estimado como del tiempo de ciclo resultante.

A.2.2.4.2 Fases de la metodología

La Figura A.10 muestra los pasos seguidos en este trabajo, numerados y resaltados en negrita y cursiva, y que se combinarían con aquellos relativos a los de un proyecto de una solución automatizada.

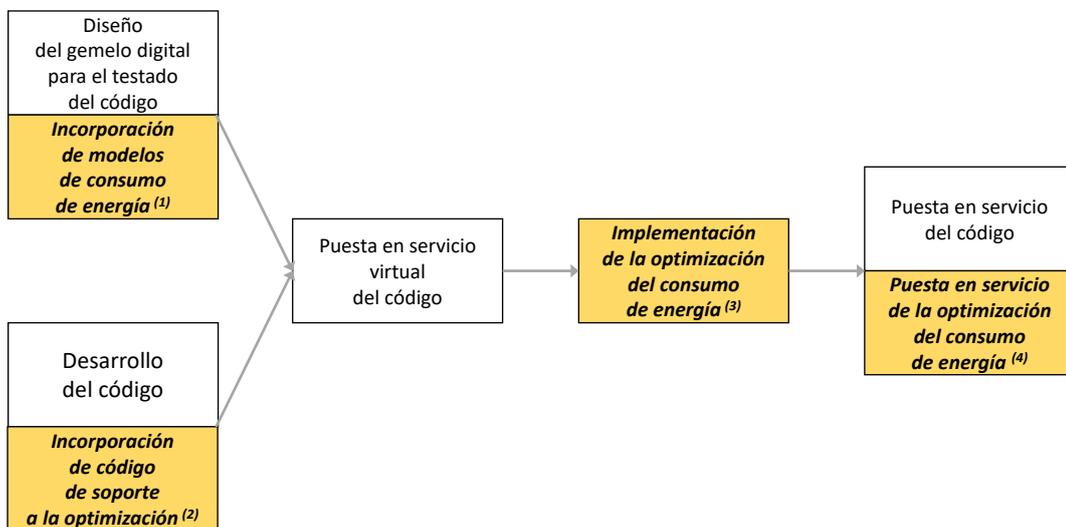


FIGURA A.10: Fases de la metodología.

0. Implementación del código de control en respuesta a las especificaciones de funcionamiento.
1. Incorporación de la estimación del consumo energético.
Caracterización de modelos nivel componente y/o uso de aquellos que incluyan esta funcionalidad.
2. Incorporación de código y variables de soporte a la optimización energética.

En función de la aplicación pueden requerirse variables adicionales específicas como la detección de inicio y fin de secuencia de funcionamiento a partir de otras variables, la duración del ciclo, o el consumo del último completado, entre otras.

La integración de estas variables en el marco para VC puede realizarse por dos vías. Para aquellas calculadas/escritas por el PLC, se agrega el código necesario. Las calculadas/escritas por el DT serán compartidas con el DT en términos similares a las señales de E/S.

3. Implementación de la optimización del consumo energético.

Se configuran los tres agentes que participan en el experimento como sigue: el PLC, servidor OPC UA, ejecutará continuamente el programa de control; el DT, cliente OPC UA, emulará el proceso con la frecuencia de entrada de material que se desee y compartirá variables con el PLC, tanto de E/S como adicionales; el OM, cliente OPC UA, ejecutará un algoritmo con el flujo de programa descrito en la siguiente sección.

4. Puesta en servicio de la optimización energética.

Aplicación, en el sistema industrial automatizado real, de los parámetros operativos que hacen posible el funcionamiento requerido en cuanto a especificaciones y ritmos de trabajo con el menor gasto.

A.2.2.4.3 Algoritmo

El algoritmo propuesto para estimación del gasto energético del sistema automatizado a partir de la combinación de parámetros analizada tiene este flujo de programa, en primera versión para la realización de ensayos aislados y no de manera automática:

1. Conexión del OM como cliente OPC UA al servidor del dispositivo de control.
2. Definición del acceso a variables clave para su uso por parte del algoritmo, tanto para escritura (parámetros, consignas de velocidad) en el DT como lectura (eventos y consumos) desde el DT.
3. Introducción, por teclado, del número de ciclos a completar y de la combinación de parámetros (consignas de velocidad) a aplicar.
4. Ejecución del ensayo, consistente en el número de ciclos consecutivos requeridos, que siguen esta secuencia:
 - Sin que haya ciclo de trabajo en ejecución, lectura de consumo acumulado.
 - Escritura de parámetros.
 - Detección de inicio de ciclo.
 - Detección de fin de ciclo.
 - Lectura de consumo acumulado.
 - Cálculo y registro del consumo parcial (diferencia entre las dos lecturas realizadas).
 - Lectura de la duración del ciclo.

- Registro de la duración del ciclo.

5. Desconexión del cliente OPC UA.

A partir del registro, para cada ensayo, de la combinación de consignas de velocidad aplicada, la duración de los ciclos ejecutados y el consumo energético estimado, se determinarían los parámetros que menor gasto cumplan con el tiempo de ciclo de trabajo requerido. Estos serían aplicados en la puesta en servicio del sistema físico, en caso de disponer del mismo.

A.2.2.5. Cierre del experimento

En lo que se refiere a este primer experimento, se han realizado, en el marco descrito, ensayos aislados consistentes en la escritura de parámetros en memoria del PLC y la ejecución de una serie de ciclos de trabajo consecutivos cuya duración se mide en el dispositivo y cuyo consumo energético se estima en el DT.

Se ha contado con el apoyo de una interfaz de usuario. La Figura A.11 muestra la entrada de datos para un ensayo, consistente en definir la combinación de consignas de velocidad a aplicar y el número de ciclos a analizar.

```
Enter setpoint for conveyor belt 1 (%): 95
Enter setpoint for conveyor belt 2 (%): 92
Enter number of cycles to execute: 2

---> Setpoint 1 (%): 95 / Setpoint 2 (%): 92 / Cycles: 2
```

FIGURA A.11: Configuración de un test en la interfaz de usuario.

A.2.3. Resultados

Cada test, consistente en los ciclos de trabajo consecutivos que se desee, tiene como resultados los gastos energéticos (ver Figura A.12) y la duración de cada uno de ellos.

```
Cycle 1 consumption: 0.005493327975273132 kWh
Cycle 2 consumption: 0.005493335425853729 kWh
```

FIGURA A.12: Resultados de un test, consistente en dos ciclos, en la interfaz de usuario: consumos estimados.

Los datos de entrada y salida de cada test aislado son registrados para su posterior análisis y elección de las consignas más adecuadas. A partir de la comparación entre pruebas puede determinarse la combinación de parámetros para la cual se cumpla tanto el criterio de consumo (el más bajo posible) como el del ritmo productivo (tiempo de ciclo siempre inferior al marcado por requerimientos de fabricación).

A.2.4. Discusión

El trabajo realizado y el resultado obtenido llevan a una consideración principal en comparación con estudios previos en este campo. Se trata de el hecho de poder estimar consumos energéticos en un DT y no fuera del emulador, merced al modelado nivel componente.

Se considera haber dado respuesta a las RQs de esta forma:

RQ09

¿Es posible dotar a un DT de la estimación de consumos de receptores en modelos nivel componente?

Sí. Existen herramientas de emulación de sistemas industriales automatizados que incluyen, a partir del modelado nivel componente, la funcionalidad, editable además, de estimar consumos energéticos en receptores.

RQ10

¿Puede un DT dar soporte al ajuste de parámetros operativos energéticamente óptimos a partir de la escritura de estos en el PLC a través de OPC UA, y la estimación de consumos de receptores en modelos nivel componente, para los ratios productivos deseados?

Si. Existen herramientas que posibilitan el acceso a variables a través de dicho estándar. Se puede realizar la escritura de los parámetros a testar, y la lectura de los consumos energéticos estimados para las condiciones de trabajo asociadas, así como detectar eventos en el proceso emulado para delimitar el inicio y el final de cada ciclo de funcionamiento. De esta manera se puede dar soporte a la aplicación de técnicas de optimización.

Este experimento ha tenido por objetivo tener una primera toma de contacto con el modelado nivel componente, concretamente para dar soporte a la fase de VC de manera que se puede, tras validar programas de control, ajustar velocidades de motores. Las siguientes pueden ser limitaciones de este trabajo preliminar:

- La aplicación es eminentemente manual, la cual requiere interacción con la persona usuaria para cada serie de pruebas a realizar.
- Esto la convierte en una herramienta más didáctica, para formación de alumnado más allá de la programación básica de PLCs, que con aplicación en entorno industrial.
- Por contra, la propuesta puede ser considerada como válida en lo que se refiere a la definición de las fases del trabajo. El reto consistiría en automatizar la parte del algoritmo.

A.3. Trabajo relacionado

La lectura/escritura de variables del/en el PLC a través del estándar OPC UA, tales como los consumos de receptores estimados en modelos nivel componente en el DT/los parámetros tales como consignas de velocidad para VSDs, puede apoyar, en una siguiente fase del trabajo, la aplicación de técnicas de optimización automática.

A.4. Conclusión y trabajo futuro

Más allá del habitual intercambio de señales entre dispositivo de control y sistema emulado para el testado de código, el acceso desde una aplicación externa a variables del primero, estén o no compartidas con el segundo, hace posible ensayos que pueden apoyar, por ejemplo, la determinación de las condiciones de trabajo idóneas

en términos de consumo energético. Se escriben consignas de velocidad, se ejecutan ciclos de funcionamiento y se lee el consumo estimado. Esto posibilita dar con el conjunto de consignas que da respuesta con el menor gasto energético a los ratios productivos requeridos sin necesidad de disponer del equipamiento físico montado y el sistema de control puesto en servicio.

Nótese que se puede dar soporte a una optimización en los términos descritos en cualquier proyecto de automatización que incluya tanto dispositivos de control como herramientas de emulación con conectividad **OPC UA**. Estas últimas, además, deben hacer posible el modelado nivel componente y la estimación de corrientes y consumos energéticos. Además, el **OM** utilizado es adaptable a otras aplicaciones y especificaciones.

La aportación del **DT** no se ha limitado a su uso más habitual, es decir, la gestión de señales de sensores y a preactuadores o actuadores para el testado de código de **PLC** y su **VC** en modelos nivel E/S. El soporte de la emulación nivel componente a la parametrización de un sistema industrial automatizado hace posible evitar el ajuste empírico, con los inconvenientes que ello conlleva, de consignas de velocidad tras la fase de puesta en servicio del mismo.

Una vez se apliquen automáticamente técnicas de optimización se pretende afinar lo propuesto en tres frentes: dotar a los modelos energéticos de una mayor precisión, realizar aplicaciones de mayor complejidad, y comparar los resultados de la emulación con los de una implementación completa en un sistema físico.