

Desarrollo de gemelo digital: evaluación de herramientas industriales y metodología

González, P.*, Gómez, J., Gómez-Jiménez, J., Parra, J.K., Escaño, J.M.

Dpto. Ingeniería de Sistemas y Automática. Universidad de Sevilla, 41092, Sevilla

To cite this article: González, P., Gómez, J., Gómez, J., Parra, J. K., Escaño, J. M. 2023. Development of a digital twin: evaluation of industrial software and methodology. XLIV Jornadas de Automática, 150-155. <https://doi.org/10.17979/spudc.9788497498609.150>

Resumen

En el presente artículo se tiene como objetivo la evaluación de algunas herramientas industriales para la creación de gemelos digitales, así como exponer una metodología que se ha seguido para implementar un gemelo digital de una célula de fabricación flexible ubicada en el laboratorio del Departamento de Ingeniería de Sistemas y Automática de la Universidad de Sevilla.

Palabras clave: Gemelos digitales para la fabricación, Industria 4.0, Integración humano-automatización, Aprendizaje automático en modelización, predicción, control y automatización.

Development of a digital twin: evaluation of industrial software and methodology

Abstract

The objective of this paper is to evaluate some industrial tools for the creation of digital twins, as well as to expose a methodology that has been followed to implement a digital twin of a flexible manufacturing cell located in the laboratory of the Department of Systems Engineering and Automatic Control of the University of Seville.

Keywords: Digital twins for manufacturing, Industry 4.0, Human-automation integration, Machine learning in modelling, prediction, control and automation.

1. Introducción

Siguiendo la línea de continua evolución y desarrollo de la tecnología que propone la industria 4.0, surge el término de gemelo digital, también conocido como *digital twin* (DT) en inglés. Un DT obtiene gran cantidad de datos, ya que es capaz de integrar tanto el espacio físico como el virtual, y posteriormente, dichos datos se pueden emplear para mejorar el rendimiento de los procesos industriales, el desarrollo de productos, la planificación de la producción, etc. (Qi and Tao, Enero 2018).

El concepto de DT fue propuesto por Michael Grieves en 2003. En este primer modelo, la estructura del DT era simple: el sistema físico en el espacio real, el producto virtual en un entorno digital y la conexión de datos e información que permitan la comunicación entre ambos componentes. Ya en 2012, un año más tarde de la publicación del primer artículo acerca de los DT (Tuegel et al., 2011), la NASA redefine el concepto

de DT como una simulación probabilística y de alta fidelidad, con múltiples físicas y escalas, que refleja el estado de su correspondiente modelo físico, basándose en datos históricos, datos de sensores en tiempo real y datos físicos del propio modelo (Glaessgen and Stargel, 2012).

En la actualidad, la estructura de DT ha pasado de tener tres dimensiones a cinco: la entidad física, su modelo virtual, los datos que se obtienen de ambos, los servicios que prestan los distintos componentes y las conexiones que se dan para que todos los campos interactúen de forma dinámica entre sí. Son numerosas las distintas aplicaciones industriales de los DT que se han recogido a través de publicaciones, patentes, etc. Muchas de ellas se pueden englobar en torno al ciclo de vida de un producto y a sus distintas fases: diseño, fabricación, operación y mantenimiento. Originalmente, los DT se usaron por primera vez en la industria aeroespacial, para la previsión y manteni-

*Autor para correspondencia: pgonzalez7@us.es
Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

miento de aviones (Tao et al., 2019).

Sin embargo, el trabajo que se presenta en este artículo se centra más en la etapa de producción dentro del ciclo de vida de un producto. Mediante el empleo de DT, se puede optimizar la producción de los distintos productos de una fábrica, prediciendo el comportamiento de los procesos, haciendo simulaciones en paralelo, monitorizando en tiempo real,... Son realmente numerosas las ventajas que se pueden obtener con el uso de DT en la industria en esta etapa de producción del producto si el DT está bien diseñado e implementado.

El objetivo principal de este trabajo es mostrar la evaluación que se ha realizado de distintas herramientas industriales para la creación de DT, dando a conocer las ventajas e inconvenientes de cada una de ellas. Además, se muestra el procedimiento que se ha seguido para la elaboración del DT de una célula de fabricación flexible, ubicada en el laboratorio del departamento de Ingeniería de Automática y Sistemas de la Universidad de Sevilla, el cual ha sido dentro del proyecto europeo *Digital intelligence for collaborative ENergy management in Manufacturing* (DENiM), cuyo propósito es el control y la optimización de la energía en las instalaciones de producción, procesos y máquinas, mediante el uso, entre otras herramientas, de DT (DENiM, 2020).

La organización del artículo es la siguiente: en la sección 2 se muestra las herramientas industriales que se han evaluado para la creación de DT, en la sección 3 se explica el procedimiento que se ha seguido para la elaboración del DT de la célula de fabricación flexible y en la última sección se muestran las conclusiones obtenidas, así como trabajos futuros.

2. Evaluación de herramientas industriales

Para el presente artículo se han evaluado cuatro herramientas industriales para la creación de DT: *Siemens Tecnomatix Plant Simulation*, *NX MDC*, *Visual Components* y *Unity 3D*. No son las únicas, ya que existen otras muchas empresas que debido al auge de los DT han ido desarrollando programas para su implementación. Para su evaluación, se ha descargado la herramienta y se ha procedido a explorar sus distintas funcionalidades. A continuación, se muestra una breve descripción de cada una de ellas, así como las ventajas e inconvenientes que se han encontrado en su uso.

2.1. Tecnomatix Plant Simulation

El conjunto de soluciones *Tecnomatix*, desarrollado por la empresa alemana *Siemens* está orientado a conseguir la sincronización de ingeniería de los productos y de fabricación con las operaciones y la producción para así maximizar la eficacia de esa producción (Siemens, 2023). Dentro de este conjunto, se ha evaluado *Tecnomatix Plant Simulation*, que es una herramienta de simulación y programación de eventos discretos (*Discrete-Event Simulation*, DES) que permite modelar, simular, explorar y optimizar sistemas de producción y logística. Es una herramienta muy versátil, ya que permite la simulación de sistemas complejos, como por ejemplo, la simulación de la producción de una fábrica completa, incluyendo la logística y el transporte de materiales, como se puede ver en la figura 1. Además, permite la simulación de sistemas de producción y logística en 3D, lo que permite una mejor visualización de los procesos. Por

último, permite la integración con otras herramientas de *Tecnomatix*, como *Process Simulate*, *RobotExpert* y *Plant Design*.

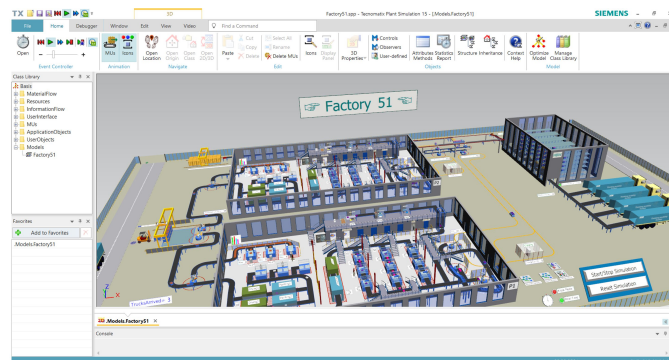


Figura 1: Ejemplo de una fábrica completa en Tecnomatix Plant Simulation

Como inconvenientes, al ser un software que usa el método DES, no está tan orientado hacia la exactitud del modelado 3D o la detección de colisiones. Además, no incluye motor físico y, por tanto, no se pueden obtener datos a bajo nivel de los componentes del sistema.

2.2. NX MDC

El software NX es una potente herramienta de diseño, simulación y fabricación desarrollada por la empresa *Siemens*. Dentro de este software, se ha evaluado la extensión de *Mechatronic Concept Designer* (MCD), que ofrece grandes posibilidades para la creación de DT. Permite el trabajo simultáneo de la ingeniería mecánica, la eléctrica y de automatización, con un lenguaje común que facilita la integración al final del proceso de diseño. Además, posibilita la integración con otras aplicaciones de NX, como *NX CAM*, *NX CAD* y *NX Flow*.

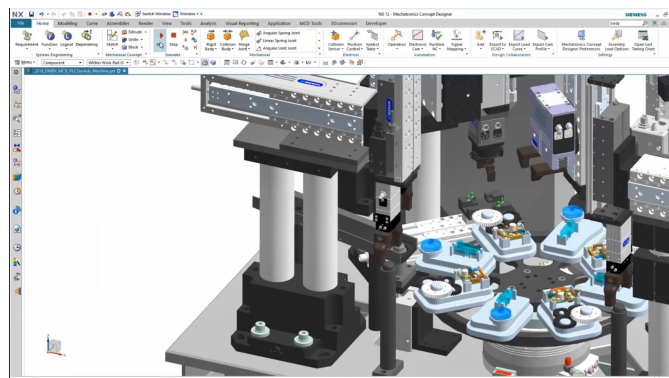


Figura 2: Ejemplo de un proceso en NX MCD

Con más detalle en el modelado 3D que *Tecnomatix Plant Simulation*, como se puede ver en 2, posibilita el diseño y la simulación de procesos, y, gracias a su motor físico, permite ir detectando funcionamientos erróneos, colisiones, caídas de objetos, etc. Presenta una gran facilidad para la comunicación con programas externos, como *MATLAB*, servidores, PLCs, etc. Aparecen como grandes inconvenientes el elevado coste de la licencia, así como la dificultad de su aprendizaje.

2.3. Visual Components

Esta herramienta, diseñada por la empresa finlandesa del mismo nombre, está orientada a la simulación 3D para la producción a nivel profesional, con una plataforma muy visual y flexible, con presencia en sectores como la automoción, la robótica, la logística, la electrónica, la maquinaria,... (Visual-Components, 2023). El objetivo principal es la simulación de líneas de producción, incluyendo vehículos autoguiados, robots, etc, y la obtención de datos en tiempo real para su posterior análisis.

Posee una librería de modelos virtuales con más de 2700 componentes de las principales marcas del mercado (ABB, KUKA, Toshiba, Schneider Electric,...). Se pueden importar y exportar archivos en numerosos formatos, y tiene una gran facilidad de conexión mediante OPC UA, WinMOD Net, SIMIT y Siemens S7. Aspecto nuevo con respecto a las anteriores herramientas es el acceso a diversas interfaces de programación de aplicaciones (API) de Python (Figura 3) y .NET, entre otras, mediante los que se puede personalizar la propia aplicación, crear pluggins, generar distintos KPIs, etc.

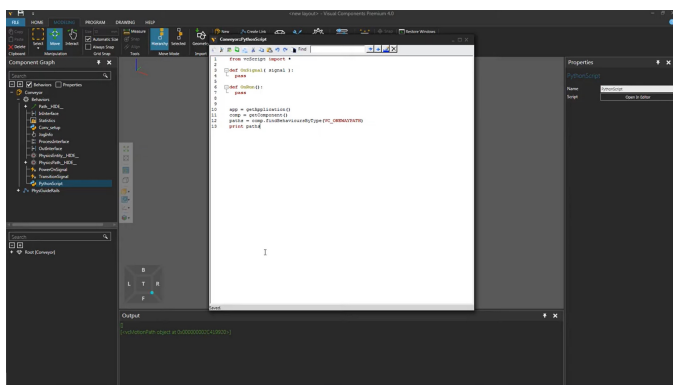


Figura 3: Ejemplo API Python en Visual

2.4. Unity 3D

Unity 3D es un software diseñado en un principio para la creación y el desarrollo de videojuegos pero que en los últimos años se ha ido extendiendo a otros ámbitos, como la ingeniería, la fabricación, etc. Tiene muchas ventajas con respecto a las herramientas vistas anteriormente ya que es un software de código abierto con motor físico, capaz de obtener datos dinámicos, físicos, etc. a bajo nivel. Además, es un software gratuito si su finalidad es educativa o de investigación.

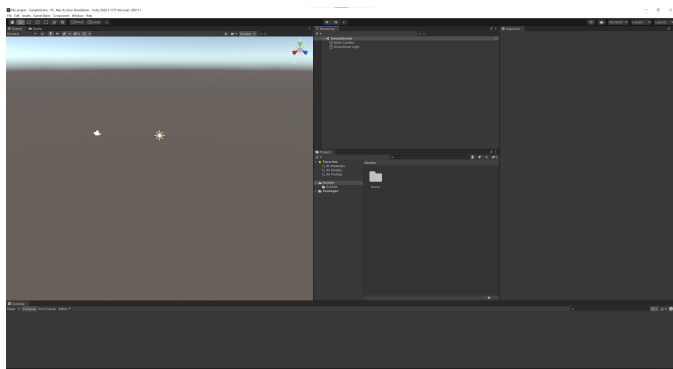


Figura 4: Interfaz de usuario de Unity 3D

Su interfaz de usuario, aunque intuitiva, como se ve en 4, requiere un aprendizaje costoso al principio, así como conocimientos previos de programación. Además, no posee una librería de componentes, por lo que es necesario crearlos desde cero, lo que requiere un gran esfuerzo y tiempo, que al mismo tiempo puede ser una ventaja, ya que se pueden crear componentes a medida de las necesidades del usuario en un software de diseño 3D.

2.5. Comparativa

Se ha realizado una tabla comparativa entre las herramientas evaluadas (ver tabla 1) en la que se muestran las principales características de cada una de ellas

Finalmente, se ha decidido utilizar *Unity 3D*, ya que, como se en la tabla 1, se puede modelar en 3D y modificar elementos a bajo nivel gracias a que es una plataforma abierta. Con ello, se puede reproducir de forma más fiel a la realidad el comportamiento de la célula de fabricación. Así mismo, también ha pesado mucho en la elección, además de las dos características ya mencionadas, la accesibilidad del producto frente a las otras herramientas analizadas, debido a su facilidad de descarga y a su precio gratuito. Estas ventajas se han impuesto así a los inconvenientes de no tener una librería de modelos virtuales, como *Visual Components*, o de que su aprendizaje no sea tan intuitivo como en *Tecnomatix Plant Simulation*, cuyo impacto ha sido menor a la hora de elegir *Unity 3D*, ya que se ha buscado más la exactitud del modelo a la realidad.

3. Metodología para el desarrollo de un gemelo digital

Definido ya el entorno de trabajo, en esta sección se desarrolla la metodología para la creación del DT de la célula de fabricación flexible de la figura 5. La célula tiene diversos componentes, aunque actualmente solo se han integrado en el DT las cuatro cintas transportadoras, el apilador de bandejas y el cuadro de control, como se verá más adelante.



Figura 5: Vista general de la planta física

Como indican Gómez Jiménez (2021a) y Gómez Jiménez (2021b), se ha definido una metodología específica, que se ha empleado en el desarrollo del DT. Dicha metodología se compone de los siguientes pasos:

Tabla 1: Comparación de las características de las herramientas evaluadas.

Software evaluado	<i>Tecnomatix Plant Simulation</i>	<i>NX MCD</i>	<i>Visual Components</i>	<i>Unity 3D</i>
Datos en tiempo real	Sí	Sí	Sí	Sí
Datos a bajo nivel	-	-	-	Sí
Librería de componentes	Solo componentes genéricos	-	Gran variedad de marcas comerciales	-
Diseño 3D	Sí	Sí	Sí	Sí
Modelado 3D	-	-	-	Sí
Detección de colisiones	-	Sí	Sí	Sí
Motor físico	-	Motor propio	NVIDIA Physx	NVIDIA Physx
Realidad virtual	-	-	Sí	Sí
Importación de archivos	-	Pocos tipos	Muchos tipos	Muchos tipos
Exportación de archivos	Sí	Sí	Sí	Sí
Conectividad	OPC UA	MATLAB, OPC UA, TCP/UDP y Siemens S7	OPC UA, Siemens S7, SIMIT y WinMOD	MATLAB, OPC UA, TCP, etc.
Plataforma abierta	-	-	API Python y .NET	C#
Fácil aprendizaje	Sí	-	Sí	-
Ayuda en la red	Sí	-	Sí	Sí
Precio	Alto	Alto	Medio	Gratis para educación

1. Arquitectura interna
2. Modelado visual
3. Modelado analítico
4. Autoaprendizaje
5. Conectividad
6. Servicios

Las características principales de esta metodología son la precisión y la escalabilidad. La precisión se consigue mediante la creación de modelos conformes a la realidad, así como de dotar al DT de la capacidad de autoaprendizaje. Así mismo, la escalabilidad permite que, ante modificaciones en el entorno real de la célula, el DT se puede adaptar a dichos cambios sin necesidad de realizar grandes cambios en la programación del mismo.

A continuación, se hará un breve explicación de cada uno de estos pasos.

3.1. Arquitectura interna

La arquitectura interna del DT es muy similar a una arquitectura de software descentralizada, esto es, que cada tarea la realiza un único módulo distinto. De esta forma, aunque la programación sea más compleja, se consigue una mayor escalabilidad en todo el gemelo. Sin embargo, la estructura no es del todo descentralizada, porque existe un módulo principal, el *Main*, que se encarga de inicializar todos los modelos del DT, así como la comunicación, el aprendizaje y los distintos servicios entre ellos. La arquitectura interna queda definida como se ve en la figura 6. El núcleo central de la estructura interna serían el modelo analítico, junto con el módulo de entrenamiento para el autoaprendizaje, todavía en desarrollo, así como un componente de comunicación, *GateComponent*, que transmite los datos de actuadores, sensores, etc, a los distintos servicios. En paralelo a este núcleo central, se encuentra el módulo principal y el módulo de inicialización, encargados de inicializar todo el gemelo y controlarlo; el módulo de análisis de datos, también en

proceso de desarrollo; y el módulo que se encarga de la comunicación con el sistema real y los distintos servicios, como la automatización, realidad virtual, etc. Por último, el módulo de visualización, que se encarga de mostrar el DT en una interfaz gráfica.

3.2. Modelado visual

Como ya se ha comentado, para un primer acercamiento al DT, se decidió modelar únicamente las cintas transportadoras y los elementos para su funcionamiento (sensores, actuadores, etc). Más tarde, se modeló también el apilador de bandejas y el cuadro de mandos (Olivar et al., 2022).

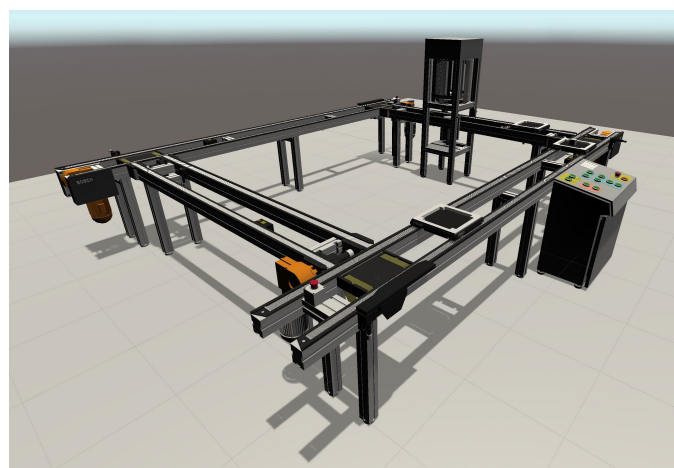


Figura 7: Modelo 3D de la célula de fabricación

El modelo 3D de la célula de fabricación se ha elaborado con *Cinema 4D*, un software orientado al modelado 3D, animación y renderizado, de fácil aprendizaje. El resultado de este modelado se puede ver en la figura 7, donde se muestra el modelo ya exportado a *Unity 3D*. Este modelo final presenta algunas diferencias con el modelo 3D diseñado en *Cinema 4D*, ya al

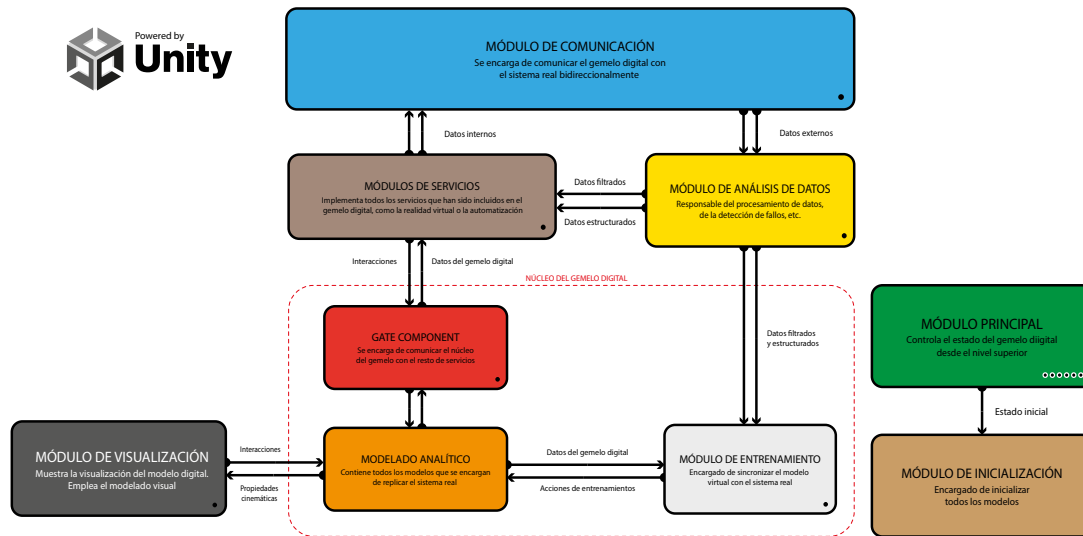


Figura 6: Arquitectura interna del DT

exportarlo, se perdían mucha información relacionada con las texturas. Sin embargo, al tener un alto nivel de detalle el modelado, se decidieron no añadir unas nuevas texturas al modelo de *Unity 3D*.

3.3. Modelado analítico

Una vez creado el modelo visual, el siguiente paso para la creación del DT es el modelado analítico. Comienza aquí la programación del comportamiento de cada elemento del modelo 3D, de manera que dicho elemento se asemeje lo máximo posible a la realidad. Para ello, se ha utilizado el motor físico de *Unity 3D*, encargado del cálculo de efectos cinemáticos y dinámicos básicos, y que facilita el desarrollo del gemelo, gestionando efectos como el rozamiento o la gravedad de los objetos.

A los distintos objetos creados en el modelo 3D se le han ido añadiendo componentes necesarios para crear su parte física. Estos componentes van desde los más básicos, como el componente *Rigidbody*, que añade la física básica a un objeto, o el *Collider*, que crea el volumen que delimita un objeto para que pueda colisionar con otros objetos físicos, hasta algunos más complejos, como el componente *Hinge Joint*, que permite crear articulaciones entre objetos. A la hora de definir estos elementos físicos, se ha intentado simplificarlos, despreciando algunos efectos físicos que no eran tan relevantes para la simulación, como las deformaciones sufridas por los objetos al colisionar, o las fuerzas de sujeción de los tornillos de la cinta.

Se han desarrollado los modelos analíticos de: las cintas, los pistones de cambio de cinta, los pistones de parada de cada cinta y el apilador de bandejas, los sensores inductivos, y otros objetos como bandejas, tanque de aire (no tiene modelo visual), etc. Los valores que devuelven estos actuadores, sensores y objetos se pasan al resto del DT gracias al módulo *GateComponent*, que permite obtener los distintos valores y modificarlos cuando sea preciso. La programación de los distintos modelos está realizada para que solo se pueda acceder a sus datos a través de este

componente, de manera que se consigue una mayor seguridad en el DT.

3.4. Autoaprendizaje

La idea es que los modelos analíticos sean dinámicos, es decir, que puedan amoldarse al sistema real. Para ello, se ha desarrollado un módulo de autoaprendizaje, que se encarga de recoger los datos de los sensores y actuadores del sistema real, y modificar los modelos analíticos para que se asemejen lo máximo posible a la realidad. Primero, con un tipo de entrenamiento *offline*, para cuando se inserte un nuevo modelo al DT, y más tarde, con un entrenamiento *online*, para cuando el DT esté en funcionamiento y se vayan desarrollando cambios en el sistema real.

Este módulo se encuentra en desarrollo, ya que se está explorando la idea de crear una “red genérica”, la cual identifique cualquier sistema, lo entrene y después pueda evaluarse y compararse con el error máximo.

3.5. Conectividad

Una de las partes más importantes del DT, ya que es la que une el sistema real con el modelo virtual. Actualmente, el módulo de comunicación está compuesto de un código principal, que se encarga de iniciar dos tipos de comunicaciones: una, mediante el protocolo *OPC UA*; y otra, mediante el protocolo *MODBUS TCP/IP*. Gracias a esta última, se consigue la comunicación tanto con el PLC que gobierna las cintas, como con el que controla el apilador de bandejas. Con la comunicación *OPC UA*, permite la interacción de un servidor externo que puede acceder al gemelo. Además de este código principal, existen otros códigos que incluyen todos los parámetros necesarios para llevar a cabo ambas comunicaciones, así como la conexión con el PLC.

3.6. Servicios

El último módulo del DT, pero no por ello el menos importante, ya que es el que le da la utilidad al DT. Se tienen

implementados la automatización y la realidad virtual (VR). La automatización se ha desarrollado de forma que se pueda seleccionar entre una simulación interna, solo el modelo virtual, de forma *offline*, desconectada de la realidad, y una automatización externa, que permite la conexión con el sistema real mediante el módulo de comunicación. Así mismo, la realidad virtual, permite la visualización del modelo virtual no solo en el entorno de Unity 3D, sino también en un entorno virtual, mediante unas gafas de realidad virtual, y con la posibilidad de interactuar con el modelo.

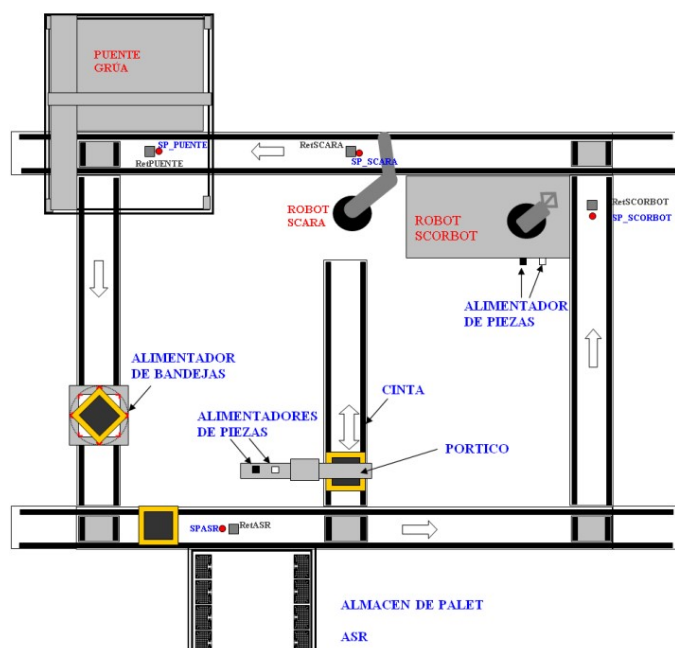


Figura 8: Esquema de la célula de fabricación

El proceso de automatización que se ha seguido es muy sencillo. Para el funcionamiento de las 4 cintas transportadoras se han creado 2 procesos distintos: uno, para el cambio de la cinta larga a la cinta corta, el cual se realiza subiendo el pistón de la esquina cuando se detecta una bandeja; y otro, para el cambio de la cinta corta a la cinta larga, que se realiza cuando la bandeja acciona el sensor de fuerza y este hace que el pistón de cambio de cinta baje. Viendo el esquema de la figura 8 (del cual, solo se han implementado las cintas, el alimentador de bandejas y sus sensores) se puede entender mejor el movimiento de las cintas.

Se está desarrollando, dentro de este bloque de servicios, una interfaz de usuario, que permita controlar mejor el uso de los distintos servicios del DT. Así mismo, también está en proceso de desarrollo unos modelos analíticos más precisos para el tratamiento de aire comprimido, de manera que se pueda implementar un servicio de optimización de la energía empleada para suministrar aire comprimido a la célula de fabricación.

4. Conclusiones y trabajos futuros

En este artículo se ha llevado a cabo una evaluación de herramientas industriales para la elaboración de gemelos digitales, dando un punto de vista objetivo sobre la elección del software empleado por el equipo de investigación *DENiM* para la elaboración del DT de la célula de fabricación flexible que se encuentra en el laboratorio de sistemas y automática de la Escuela

Técnica Superior de Ingeniería de la Universidad de Sevilla. Así mismo, se ha realizado una descripción de la metodología de trabajo empleada para la elaboración del DT, resumiendo cada uno de los puntos que se han llevado a cabo.

Como trabajo futuro, se pretende seguir desarrollando el DT, añadiendo nuevos modelos de los distintos componentes que conforman el sistema real, como el almacén de bandejas, el robot *Scorbot* o el robot *SCADA* (ver esquema 8), así como la implementación de nuevos servicios que permitan una mayor interacción con el DT, como la optimización de la energía empleada para el suministro de aire comprimido a la célula de fabricación flexible. Además, se pretende tener terminado el módulo de aprendizaje para así obtener datos de entrenamiento para la red neuronal que se pretende implementar en el DT.

Agradecimientos

Los autores agradecen a la Comisión Europea la financiación de este trabajo en el marco del proyecto *DENiM*. Este proyecto ha recibido financiación del programa de investigación e innovación Horizonte 2020 de la Unión Europea en virtud del acuerdo de subvención nº 958339. También agradecen al proyecto PID2019-104149RB-I00 financiado por MCI-N/AEI/10.13039/501100011033.

The authors want to thank the European Commission for funding this work under Project *DENiM*. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 958339. Also by Grant PID2019-104149RB-I00 funded by MCIN/AEI/10.13039/501100011033.

Referencias

- DENiM, 2020. Digital intelligence for collaborative energy management in manufacturing. URL: <https://denim-fof.eu>
- Glaessgen, E., Stargel, D., 04 2012. The digital twin paradigm for future nasa and u.s. air force vehicles. In: Proceedings of the 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference. pp. 1–14. DOI: 10.2514/6.2012-1818
- Gómez Jiménez, J., 2021a. Diseño de gemelo digital del sistema de transporte de célula de fabricación flexible.
- Gómez Jiménez, J., 2021b. Metodología para el desarrollo de gemelos digitales. aplicación a célula de fabricación flexible.
- Olivar, M. Á. R., Jiménez, J. G., Castaño, F., Escaño, J. M., Sep. 2022. Desarrollo de gemelo digital de un alimentador de bandejas e integración en entorno de realidad virtual. In: XLIII Jornadas de Automática: libro de actas: 7, 8 y 9 de septiembre de 2022, Logroño (La Rioja). Servicio de Publicaciones da UDC, pp. 591–598. URL: <https://doi.org/10.17979/spudc.9788497498418.0591> DOI: 10.17979/spudc.9788497498418.0591
- Qi, Q. L., Tao, F., Enero 2018. Digital twin and big data towards smart manufacturing and industry 4.0: 360 degree comparison. IEEE Access vol 6, pp. 3585–3593.
- Siemens, 2023. Siemens digital industries software: Tecnomatix. URL: <https://www.plm.automation.siemens.com/global/es/products/tecomatix/>
- Tao, F., Zhang, H., Liu, A., Nee, A. Y. C., Apr. 2019. Digital twin in industry: State-of-the-art. IEEE Trans. Industr. Inform. 15 (4), 2405–2415.
- Tuegel, E. J., Ingrassia, A. R., Eason, T. G., Spottswood, S. M., 2011. Reengineering aircraft structural life prediction using a digital twin. Int. J. Aerosp. Eng. 2011, 1–14.
- Visual-Components, 2023. Visual components oy. URL: <https://www.visualcomponents.com/>