

# LA INFORMÁTICA INDUSTRIAL EN LAS INGENIERÍAS INDUSTRIALES

Rogelio Mazaeda, Eusebio de la Fuente, José L. González, Eduardo J. Moya, Miguel A. García, Javier García, María J. de la Fuente, Gregorio Sainz, Smaranda P. Cristea  
[rogelio@cta.uva.es](mailto:rogelio@cta.uva.es), {efuente, jossan, edumoy, miguel, javgar, mjfuente, gresai}@eii.uva.es, smaranda@autom.uva.es.

Departamento de Sistemas y Automática, Escuela de Ingeniería Industriales, Univ. de Valladolid (UVA).

## Resumen

*El término Informática Industrial se utiliza para dar cabida a un conjunto muy diverso de disciplinas y tecnologías. El encontrar un criterio unificador que articule los diversos enfoques posibles y que, ayude tanto al docente como al alumno, en el proceso de enseñanza/aprendizaje de manera que se brinde conocimiento perdurable y al mismo tiempo actualizado, es una tarea clave. En esta contribución se discuten estas ideas relativas a la docencia de esta disciplina en las diferentes titulaciones de la Escuela de Ingenierías Industriales de la UVA, manifestando ideas de cómo ese marco conceptual puede adaptarse al contexto de las diferentes especialidades.*

**Palabras Clave:** Informática Industrial, Sistemas de tiempo real, lenguajes de programación, educación en control, arquitecturas concurrentes, sistemas distribuidos de control por computadores

## 1 INTRODUCCIÓN

El uso de los computadores digitales permea todos los aspectos de la actividad económica y social modernas. Como no podía ser de otra manera, la industria contemporánea es inconcebible sin la utilización del computador y esto ocurre desde cualquier punto de vista, desde las aplicaciones convencionales y comunes a cualquier otra esfera, a aquellas que están directamente vinculadas al control y supervisión de la actividad productiva. Incluso, con la alta escala de integración y el abaratamiento actualmente conseguidos, cada vez ocurre con más frecuencia que los productos mismos contienen la “inteligencia” demandada por el mercado en forma de controladores digitales empotrados.

Este uso industrial de la informática tiene características propias, que la distinguen de otras aplicaciones más convencionales, y que se derivan fundamentalmente del hecho de tener que interactuar no solo con el usuario humano, sino sobre todo, con el mundo físico real.

En el departamento de Ingeniería de Sistemas y Automática (DISA) de la Universidad de Valladolid (UVA), desde hace tres años se ha creado un grupo de trabajo interesado en la docencia de esta disciplina

que se ha dado en llamar Informática Industrial. Como resultado de esa labor se han llegado a algunos consensos sobre cómo abordar los estudios de una materia que es muy amplia, que no está siempre bien definida, que exige la precisión propia de la programación de ordenadores y que lidia con una tecnología que es eminentemente volátil. Estas ideas, que fueron perfiladas para ser aplicadas a la titulación de Grado en Ingeniería en Electrónica Industrial y Automática (GIEIA), han sido plasmadas en los documentos de dos Proyectos de Innovación Docente anteriores, y en sendos artículos [1],[2]. En la presente contribución ampliamos la discusión a todas las titulaciones en la que el DISA tiene docencia, en el espíritu de experiencias que, con las mismas motivaciones, se han desarrollado por otros grupos de trabajo, como la reportada [3] de la Universidad de Almería.

El artículo se estructura como sigue: en la sección 2 se hace un breve recorrido por la disciplina, en la sección 3 se establecen algunos principios que han guiado la elección de los temas a impartir, mientras que en la sección 4 se establece y comentan los temas generales identificados, mientras que en la sección 5 se discute la aplicación de estas ideas al estudio de la disciplina en las diferentes titulaciones de Grado en Ingeniería de la UVA.

## 2 LA INFORMÁTICA INDUSTRIAL

### 2.1 ALCANCE DE LA DISCIPLINA

La industria contemporánea sería inconcebible sin los computadores. La informática condiciona todas las actividades de la industria. Es pieza fundamental en el manejo eficiente de cada una de las etapas del ciclo de vida de un producto (PLM: *product lifetime management*), que incluye: la concepción, el diseño, la puesta en producción e incluso la inevitable obsolescencia del producto. Aunque el concepto englobado en las siglas PLM es mucho más amplio, el elemento informático que mejor lo representa son las herramientas de diseño, ingeniería y fabricación asistidas por ordenador (CAD/CAE/CAM).

Hay que tener en cuenta que el producto mismo puede hospedar uno o varios computadores embebidos, práctica que es cada vez más frecuente debido al desarrollo tecnológico que permite el

abaratamiento y la miniaturización de estos dispositivos electrónicos.

Una vez puesta en marcha la fabricación industrial del o los productos de que se trate, y también estrechamente vinculadas a las herramientas PLM, la informática participa a todos los niveles jerárquicos en que suele concebirse la fábrica actual, tal y como se representa en la conocida pirámide CIM (fabricación integrada por computadora), posteriormente codificada en el estándar ISA 95 [4] y en la más recientemente norma IEC 62264 (ver fig. 1). La pirámide exhibe, en la parte superior aquellos procesos de gerencia de alto nivel de la fábrica que tienen que ver con la gestión de los recursos instalados y la planificación a medio y largo plazo de la producción (ERP: *Enterprise Resource Planning*), teniendo en cuenta los requerimientos de los clientes (CRM: *Client Relationship Management*), y la eficacia de las relaciones con otras empresas para mantener la cadena de suministros y los inventarios en niveles óptimos (SCM: *Supply Chain Management*). En la base de la pirámide CIM se tiene el control de los procesos de fabricación a nivel de campo, actividad que desde hace décadas se lleva a cabo utilizando computadores más o menos especializados, conectados en red. Aquí los requerimientos temporales son más exigentes. En las posiciones intermedias se tiene, justo encima del nivel de interacción directa con el proceso físico, una capa de sistemas de control avanzado y supervisión y, ocupando una posición intermedia de enlace con la capa de gestión, se localiza el llamado nivel MES, que convierte los objetivos a largo plazo de la capa de gestión, en órdenes concretas para las capas inferiores.

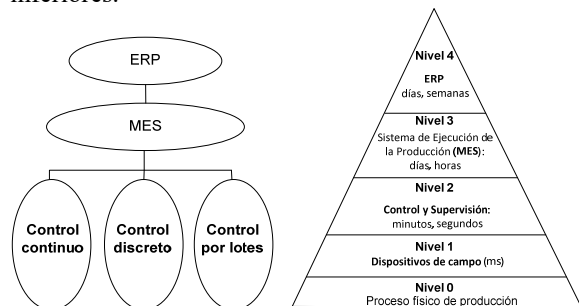


Figura 1: La pirámide CIM

Resumiendo, los tres grandes campos brevemente esbozados, que podríamos clasificar, con todas las precauciones de rigor, como relativos al diseño del producto, integración en el producto mismo y la gerencia de la fábrica a los diferentes niveles jerárquicos, constituyen ejemplos no triviales de aplicación de los computadores a la industria o de “Informática Industrial”. A la hora de definir los contenidos propios de esta disciplina, cualquiera de los enfoques mencionados es legítimo.

## 2.2 ENFOQUE ELEGIDO

En esta contribución, buscando acotar, aunque sea mínimamente, el espectro de los temas a tratar y también procurando la vinculación con los relacionados al control y la automatización, elegimos considerar como temas de interés aquellos que describen la aplicación de los computadores que cumplen con los siguientes rasgos:

1. Tratan fundamentalmente (aunque no exclusivamente) con el mundo físico.
2. Debe reaccionar a diferentes eventos que pueden ocurrir, en principio, simultáneamente y en un orden que no puede ser anticipado.
3. El plazo de reacción del sistema informático al evento está acotado.
4. Los elementos a controlar o supervisar pueden estar dispersos geográficamente.
5. El malfuncionamiento del sistema informático puede acarrear consecuencias muy graves en términos económicos o de daños personales.

Si nos atenemos a los principios mencionados, que no pretenden ser novedosos, excluiríamos de nuestro interés las aplicaciones de diseño de tipo CAD/CAE, como por ejemplo, sistemas de diseño mecánico, químico o eléctrico, entre otros, que deberían ser abordadas en el contexto de las ingenierías específicas correspondientes. Aquí habría que hacer la salvedad importante de que los sistemas de diseño de soluciones para productos “inteligentes” empotradas, sí que serían motivo de atención legítima.

A pesar de la anterior demarcación del ámbito de estudio, el enfoque de la disciplina continúa presentando muchas dificultades inherentes, que ya fueron adelantadas en la introducción y que constituyen un reto para su estudio. La informática industrial concebida en los términos antes mencionados, constituye un campo de actividad multidisciplinar que bebe de las fuentes de dos importantes cuerpos teóricos bien asentados: como son la informática o ciencias de la computación y la teoría de sistemas y el control automático. Esta situación intermedia, que indudablemente constituye una señal de la importancia del enfoque que representa, puede ser una desventaja a la hora de afrontar su estudio, debido a la amplitud y diversidad de los temas a tratar, que le otorga un cierto carácter disperso. Dificultades añadidas, se encuentra en la complejidad intrínseca de las fuentes de la se nutre. Tanto la informática como la teoría de control tienen un importante basamento teórico.

En el caso de la ciencia de la computación, se tienen los importantes resultados relacionados con la jerarquía de las gramáticas formales, el análisis de la complejidad computacional y el problema de la decisión [5], además de una plétora de algoritmos profusamente estudiados [6], de diferentes

paradigmas y lenguajes de programación y de patrones de diseño [7].

La teoría de sistemas y el control automático ofrece igualmente un cuerpo teórico casi inabarcable que va desde los conceptos importantes de realimentación y estabilidad, aplicados a sistemas continuos, lineales y no lineales y toda una serie de aplicaciones avanzadas conocidas bajo los rótulos, para mencionar solo algunos, de control óptimo, robusto, adaptativo, predictivo, etc, y que hace uso de los conceptos de modelado o identificación dinámica de sistemas, de la simulación y la optimización matemática, entre muchas otras técnicas, capaces de lidiar con sistemas inciertos y que dan como resultado soluciones aplicables a todos los niveles de la pirámide de control. Para una reciente y autorizada visión de conjunto de la disciplina ver [8].

Ciertamente, la relación entre la informática y el control automático es muy estrecha y es muy rica en matices. Desde hace mucho tiempo, el principal vehículo tecnológico para implementar las soluciones de control son los computadores, ya sean de propósito general o con características específicas como los controladores lógicos programables o PLC. La inmensa mayoría de las aplicaciones de control avanzado, con algoritmos que incluyen el uso en línea de modelos e incluso la optimización matemática de los mismos, serían imposibles sin la capacidad de cálculo que aportan los ordenadores.

### 2.3 PREVISIBLE EVOLUCIÓN DE LA INFORMÁTICA EN LA INDUSTRIA

Pero la relación entre computadores y control no se limita a concebir los primeros como los dispositivos tecnológicos que mejor permiten llevar a la práctica los conceptos del control en el presente momento del desarrollo de la técnica. Por supuesto, el uso de ordenadores motivó en su momento que la teoría de control se ocupará de la necesidad de estudiar los efectos del muestreo y la cuantificación de las señales continuas para su procesamiento digital [9]. Los resultados ya clásicos relativos a sistemas de tiempo discreto se estudian a la par en pie de igualdad con el enfoque continuo original. Estudios de esta índole se continúan llevando a cabo, por ejemplo en [10], para verificar los efectos que sobre los sistemas de control realimentados y distribuidos, tiene el comportamiento de las redes informáticas. Pero en el sentido inverso, también ha existido un influencia notable, la teoría de control hoy acepta, casi en pie de igualdad, las aplicaciones reactivas dedicadas a la automatización de procesos discretos, a partir de los trabajos sobre control supervisor introducidos en [11], esfuerzos teóricos que continúan desarrollándose y que se basan en la aplicación de conceptos y modelos (sistemas de eventos discretos, máquinas de estado finitas y

similares, con sus gramáticas asociadas) de la ciencia de la computación [12].

De forma que se abre paso, un enfoque conjunto en la intersección de estas dos grandes disciplinas, enfoque del que se hace eco en el ya mencionado artículo [8] y que es adoptado por ejemplo en [13]. La confirmación de este nuevo punto de vista, lo constituye la emergencia de conceptos como el de los sistemas ciber-físicos (CPS: *cyber-physical systems*) [14], que promueve el estudio conjunto de la realidad física a controlar y de los controladores basados en computadores que se encuentran frecuentemente conectados en red, ejecutando aplicaciones reactivas distribuidas en tiempo real. En el enfoque CPS, la forma natural de modelado de estos sistemas complejos será híbrida, combinando el modelado continuo, junto a los modelos de eventos discretos. Este enfoque está en la base o muy relacionado con conceptos como el de la Industria 4.0, el de internet de las cosas (*IoT*), el uso masivo de datos de diversas fuentes (*big data*), el uso de la nube (*cloud* o *fog computing*) y el uso de redes inalámbricas también para el control [15], [16], [17]. La industria 4.0 es todavía una aspiración más que una realidad asentada, pero ya ha movilizad el interés de la academia y de la industria, y su implantación supone la adaptación del modelo representado por la pirámide ISA-95 a las nuevas exigencias, puesto que sin negarlo tajantemente, la nueva concepción que emerge estimula una arquitectura donde los diferentes actores operen de forma más horizontal y menos atados a jerarquías rígidas. Los retos a vencer en este nuevo contexto son muy considerables. Para poner solo un ejemplo, un escenario probable será la existencia de lazos de control distribuidos conectados a distancia, donde el sensor alimentado con batería enviará por red inalámbrica sus lecturas al controlador que puede estar a distancia considerable, desplegado, por ejemplo, junto con el control del actuador. En este ámbito existen dificultades a vencer relacionadas con el ancho de banda del lazo de realimentación, la posible pérdida de información, el análisis de los riesgos que toda esta incertidumbre implica, añadido a las amenazas relacionadas con la ciber-seguridad. En un contexto como el anterior, existen ventajas de ahorro de energía en el sensor, en abandonar el uso de la teoría de sistemas muestreados, con las simplificaciones teóricas que ello conlleva, para implantar un esquema que solo envía datos al controlador si la señal de proceso varía sustancialmente. De manera, que está posibilidad práctica impulsa la necesidad de tratar con sistemas basados en eventos, probablemente más eficientes pero menos estudiados [18] para el control de variables esencialmente continuas.

### 3 ALGUNAS CONSIDERACIONES GENERALES SOBRE LA

## DOCENCIA DE LA INFORMÁTICA INDUSTRIAL

La Informática Industrial, como ha sido descrita de forma inevitablemente incompleta en la sección anterior, está en el centro de este esfuerzo de renovación de la industria: lejos de ser una mera colección de tecnologías de *hardware* y *software* útiles para el control, la automatización y la gerencia de la fábrica; brinda por el contrario, además de los recursos técnicos mencionados y de constituir el elemento de integración de la industria moderna, un marco teórico unificador muy en sintonía con el actual enfoque basado en los sistemas CPS.

### 3.1 PRINCIPALES DESAFÍOS DOCENTES

Desde el punto de vista docente la dificultad no es menor y debe vencer los siguientes desafíos:

1. Importante base teórica en la intersección de la ciencia de la computación y la teoría de sistemas y el control automático.
2. La programación de computadores requiere de un muy alto nivel de compromiso por parte del programador. Es implacable con los errores sintácticos o semánticos. Supone la adquisición de habilidades que requieren tiempo y práctica. La programación de aplicaciones reactivas de tiempo real agrava de manera importante esta situación y exigen procedimientos fiables de verificación y validación.
3. Las tecnologías concretas, tanto de hardware como de software, típicamente se hacen obsoletas muy rápidamente.
4. Múltiples aplicaciones de carácter diverso, transversal y de interés, en diverso grado, para todas las titulaciones de ingeniería. La informática es importante a diversos niveles de la industria y sus temas de interés van desde las aplicaciones de control a nivel de campo, o a nivel de la automatización de dispositivos empotrados, hasta aplicaciones de carácter supervisor, como pueden ser sistemas de control avanzado, o en capas superiores de la pirámide CIM, aplicaciones de optimización económica o de ayuda a la decisión: sistemas de reconciliación de datos y optimización en tiempo real (RTO), sistemas de detección de fallos, de control estadístico de procesos, o de programación (*scheduling*) de las operaciones de la fábrica, entre muchas otras. Existen además campos específicos de aplicación con una entidad propia, como son: la robótica, la visión artificial, la mecatrónica, los vehículos guiados automáticamente (AGVs), entre otros. Cada una de estas aplicaciones requiere un conjunto muy específico de conocimientos y habilidades a diferentes niveles de abstracción.

## 4 TEMAS FUNDAMENTALES

A la luz de la discusión en las secciones previas, se han identificado un conjunto de grandes temas de los cuales extraer los contenidos concretos para impartir en la disciplina. Los temas elegidos constituyen una puesta al día de los adoptados en una contribución previa por parte de los mismos autores [1].

En la tabla 1 se numeran y describen brevemente los grandes temas en los que distribuir la docencia de la disciplina que nos ocupa.

Tabla 1: Temas

N.	Tema	Descripción
1	La informática en la industria	Pirámide CIM (ISA-95) e Industria 4.0.
2	Sistemas reactivos	El computador en el lazo de control de sistemas continuos, discretos e híbridos. Implicaciones en el hardware, Sistema operativo, lenguaje de programación.
3	Bases teóricas y modelado formal	Sistemas continuos muestreados, de eventos discretos e híbridos. Modelado.
4	Prog. concurrente	Concurrencia y paralelismo.
5	Sistemas de Tiempo Real (STR)	Conceptos y algoritmos.
6	Sistemas Distribuidos y redes industriales	Redes industriales. Determinismo, latencia y ancho de banda. Control distribuido.
7	Cíber-seguridad y seguridad funcional	Seguridad funcional y frente a ciber-ataques
8	Sistemas de control industrial	Sistemas de control industrial. PLCs, DCSs, SCADAS. Aplicaciones especiales: mecatrónica y AGVs, visión artificial.

La numeración de los temas en la tabla 1 sugiere una precedencia en la exposición temporal que resulta además inevitable. No debe, sin embargo, interpretarse como delimitando temas aislados y auto-contenidos. Por el contrario, el éxito de un curso de esta naturaleza está en encontrar un relato común unificador que viene inicialmente expresado por el contenido de los temas 1 y 2. En el tema 2 se ofrecen los conceptos centrales de sistema reactivo, de las implicaciones que estos tienen en su relación con el

mundo físico que es simultáneo y exige repuestas con plazos temporales bien determinados. Ahora bien, la existencia de esos plazos, el menor o mayor margen temporal que brinden, el hecho de si son periódicos o disparados por eventos relacionados con el comportamiento de alguna otra variable diferente al tiempo, los riesgos más o menos graves en que se incurre al violarlos, entre otras muchas cosas dependen, en gran medida, de la posición en la pirámide ISA-95 con la que se lidie o de si se trata del controlador para un sistema embebido. De manera que este tema 1 debe servir, no de mera introducción sino, junto con el tema 2, de hilo conductor y *leitmotiv* de todo el curso.

Abundando en este concepto, una aplicación del nivel MES, pongamos por caso, la programación óptima de la operación de una fábrica que debe funcionar con un plan a varios días vista, puede ser considerado un sistema de tiempo real, pero los riesgos de no cumplir el plazo exigido son menores (quizá un funcionamiento sub-óptimo durante algunas horas) que el riesgo de incumplir el de responder con una latencia excesiva ante una condición crítica que ocurra en el Nivel 1 con un controlador de campo que además debe ofrecer respuestas correctas en plazos mucho más cortos. Las redes de computación constituyen otro ejemplo muy claro: los requisitos de predictibilidad, de ancho de banda y de latencia exhibidos por el protocolo y la implementación de red utilizados será, en general diferentes, para los distintos niveles y aplicaciones: redes de campo predecibles, de baja latencia y poco tráfico en los niveles inferiores y otras más convencionales en los niveles superiores. Otro tanto se tiene con los aspectos de ciber-seguridad y seguridad funcional (tema 7) y por supuesto, las características, el tipo de lenguaje o de aplicación programables o configurables explicados en el tema 8, dependerán claramente del nivel al que nos encontramos: sistemas de tipo PLC en los niveles de control más bajos, sistemas de control distribuido (DCS) en una capa algo superior y todavía por encima sistemas supervisores tipo SCADA, bases de datos de históricos y sistemas de ayuda a la decisión basados en dichos datos.

El tema 3 cubre los pre-requisitos básicos derivados de las dos grandes disciplinas teóricas de las que se nutre la informática en la industria. El grado de profundidad con que se aborde este tema dependerá de los conocimientos previos del alumno adquiridos en el contexto de la titulación. Un breve recordatorio de las implicaciones del muestreo y la cuantificación en el controlador por ordenador de los procesos continuos, no sería nunca tiempo mal empleado. En lo que concierne al modelado formal de sistemas discretos, se puede dedicar algún tiempo a explicar diferentes alternativas que van desde la utilización de autómatas finitos (FSM: *finite state machines*), a los *state-charts* que aportan más claridad puesto que

permiten lidiar con la complejidad y contener la explosión de estados que ocurre rápidamente con las FSMs, utilizando un planteamiento jerárquico que da cabida al modelado de procesos concurrentes, o incluso una breve introducción a las redes de Petri básicas que ayudan a explicar, de una forma muy tersa e intuitiva, no sólo el modelo de la planta discreta a controlar sino el funcionamiento del propio controlador, y que es especialmente útil al entender las complejidades de la operación concurrente de las aplicaciones de control [19]. Algunas nociones de modelado híbrido serían también oportunas: máquina de estado finita donde cada estado discreto constituye un modo de operación en las que las variables continuas evolucionan según dinámicas diferentes expresadas por ecuaciones diferenciales y algebraicas (DAEs), brindarían el marco teórico adecuado para el estudio de los sistemas CPS [20].

Los temas 5, 6 y 7 deben constituir el grueso de la disciplina. Aunque no es el único modelo de computación adecuado para los sistemas reactivos, la utilización de la concurrencia, es todavía hoy, a pesar de las complejidades que impone en el diseño, en la verificación y en la validación de las aplicaciones, el paradigma preferido para emular, desde el computador, la simultaneidad real del mundo físico. La implementación de la concurrencia, impone requerimientos en el hardware, en el sistema operativo y en el lenguaje de programación en combinación con el uso de librerías especializadas. La concurrencia, en presencia de un solo núcleo de procesamiento, no busca aumentar la utilización útil de la potencia de cómputo, que disminuye inevitablemente por los necesarios cambios de contexto. En los ordenadores actuales de varios núcleos si puede darse un aumento neto de la capacidad de cómputo (paralelismo), pero la existencia de más tareas que núcleos constituye una situación típica que nos vuelve a llevar a la situación de partida, pero incrementado la complejidad del problema. En cualquier caso, la concurrencia sí que puede disminuir la latencia en la respuesta a eventos externos, periódicos o esporádicos; pero lo que es más importante, permite la búsqueda “separación de preocupaciones”: el diseñador de la aplicación puede separar, de forma lógica, las tareas de software que se encargan del tratamiento de eventos que pueden ocurrir en la realidad en orden que no es previsible. Pero fundamentalmente, la división en tareas concretas, individuales, brinda al planificador del sistema operativo en tiempo real (STR) las unidades de cómputo básicas que deben ser planificadas con precisión para cumplir los plazos especificados por el diseño. Por supuesto, pueden existir aplicaciones empotradas muy simples donde la concurrencia no aporte ningún beneficio. Por otra parte, la complejidad que se introduce en el proyecto, aumenta de forma drástica sobre todo cuando las tareas deben ser sincronizadas o comunicadas entre

sí, como es habitualmente el caso. Pueden darse situaciones anómalas como las condiciones de competencia (“*race conditions*”) o de bloqueo (“*deadlock*”) difíciles de detectar y de evitar.

Nótese que la concurrencia puede darse a nivel de hilos, que permiten un cambio de contexto más ágil y el intercambio de datos a través de la memoria común que comparten o de procesos, típicamente aislados por el sistema operativo en espacios de memoria separados y que deben recurrir a mecanismos de comunicación y sincronización, también provistos por el S.O., más lentos pero más robustos como los basados en intercambio de mensajes.

El tema 5 debe explicar las bases de los STR. Los tipos de planificadores que existen, si la planificación se realiza fuera de línea o en línea y los algoritmos más comunes, sus ventajas y desventajas. El manejo de prioridades para las distintas tareas, entre otras muchas complejidades. Para muchas aplicaciones será suficiente una aproximación a la aplicación reactiva del tipo “mejor esfuerzo” con plazos no rígidos, pero para otras será necesaria la utilización de STRs rigurosos o de “tiempo real duro”. La introducción de las prioridades añade complejidad a la programación concurrente y supone retos como la inversión de la prioridad que pueden causar situaciones de bloqueo muy sutiles.

Los computadores convencionales, tanto a nivel de hardware como de S.O., están diseñados con el propósito de dar una experiencia satisfactoria al usuario humano y no para las aplicaciones de control: la existencia de niveles de memoria cache o la implementación de la memoria virtual a nivel del S.O., implican amenazas directas al determinismo que se debe exigir a este tipo de sistemas. Los sistemas especializados para el control como PLCs y DCSs, menos versátiles y expresivos, son más predecibles para aplicaciones críticas.

El tema 6 de sistemas distribuidos presenta todas las características de los sistemas concurrentes en tiempo real basados en procesos, añadiendo las complejidades que se derivan del uso de redes informáticas.

El estudio de las aplicaciones concurrentes, en tiempo real y posiblemente distribuidas debe ser acompañado por la descripción de patrones de diseño de software y buenas prácticas de programación que serán fundamentales para garantizar una seguridad funcional adecuada, que es el contenido del tema 7, junto con las medidas necesarias para la prevención de ataques informáticos, mucho más peligrosos en un ambiente industrial.

Tiempo real, como se ha repetido muchas veces, no significa velocidad de ejecución de las tareas sino el cumplimiento de los plazos. Pero qué duda cabe, que en la medida en que esos plazos se acortan, la velocidad de cálculo y la forma en que esta se degrada con el tamaño del problema a resolver, se

vuelve cada vez más importante. En este sentido, los resultados clásicos de la ciencia de la computación sobre la complejidad de los algoritmos no deben pasar inadvertida al alumno de la disciplina (tema 3). El acudir en todo momento a soluciones de control realistas (tema 8), a la explicación de arquitecturas típicas y de casos de estudios específicos para diferentes tipos de industria brinda el contenido práctico que esta asignatura debe propiciar. También, en dependencia del caso, podrán ser motivo de discusión aplicaciones de naturaleza más particular como la mecatrónica, los AGVs y la visión artificial, siempre muy atractivos al alumno. Las aplicaciones de robótica caen perfectamente dentro de la definición de Informática Industrial que se ha adoptado, su importancia y envergadura teóricas, sin embargo, la hacen de difícil encaje en la visión generalista adoptada aquí para la disciplina.

## **5 DOCENCIA DE LA DISCIPLINA EN LA EII DE LA UVA.**

En lo que sigue se discute la estrategia adoptada para los estudios de la disciplina de II en la EII de la UVA en aquellas asignaturas a cargo del DISA.

### **5.1 PRINCIPIOS ADOPTADOS**

A los desafíos a la docencia enumerados en la sección 3.1, debe añadirse la dificultad de que deben ser afrontados en el tiempo tasado y generalmente escaso de que se dispone y de ponerlos en sintonía con los objetivos y competencias de cada titulación y en el contexto del resto de las asignaturas.

Una idea central como ya se ha esbozado, es la de utilizar como referencia o *leitmotiv* de los contenidos del curso la ubicación en el nivel correspondiente de la pirámide CIM. También se ha considerado que es fundamental que la asignatura exhiba un enfoque fundamentalmente práctico, que involucre a un nivel de abstracción u otro, la programación de computadores o al menos la configuración de aplicaciones de naturaleza industrial. Para brindar permanencia en el tiempo a los contenidos impartidos, parece importante acudir a resultados bien establecidos de la teoría de sistemas o control y la ciencia de la computación, y presentar resultados, que son clave, sobre los procesos concurrentes y los algoritmos de tiempo real y sistemas distribuidos. En los casos en que exista alternativa, se optará por la utilización de soluciones y protocolos refrendados por normas de facto o sancionadas por organismos internacionales. El carácter de la disciplina aconseja un estilo de docencia, que en consonancia con el Espacio Europeo de Educación Superior (EEES), privilegie modelos de enseñanza colaborativos, basados en proyectos y el uso fundamental del

laboratorio, reduciendo el tiempo dedicado a la tradicional clase magistral.

## 5.2 ENFOQUE EN LAS DIFERENTES TITULACIONES

En la figura 3 se grafican las asignaturas de las diferentes titulaciones susceptibles de acoger la disciplina de la Informática Industrial y aquellas que está relacionadas con ellas. El grado de Diseño Industrial no se representa porque la disciplina está ausente del mismo, aunque previsiblemente estén incluidas, de forma preferente, aplicaciones de la informática relacionadas con el ciclo de vida de los productos diseñados (PLM).

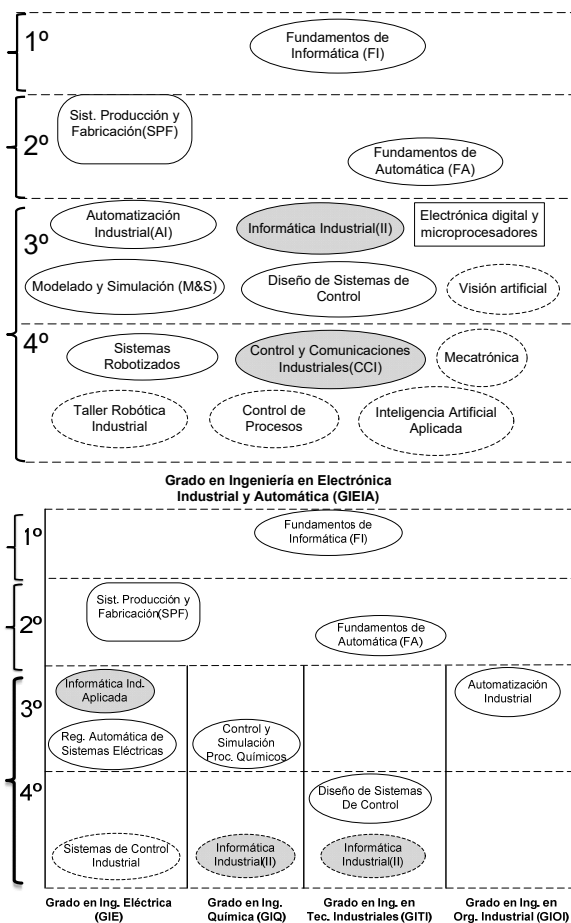


Figura 3: (a) Titulación GIEIA (b) Resto de las titulaciones (obligatorias en línea continua, opcionales en línea de puntos)

En el grado del Grado en Organización Industrial se tiene únicamente la asignatura de Automatización Industrial, como vehículo susceptible de dar contenidos relacionados con lo que nos ocupa, utilizando, por ejemplo, el estudio de los PLCs. En esta titulación, se estudiarán sin duda, aunque no impartida por el DISA, aplicaciones de la capa ERP y MES relacionadas, por ejemplo con la planificación a largo, medio y corto plazo de las operaciones

industriales y con el manejo óptimo de la línea de suministros (SCM).

Todas las titulaciones comparten asignaturas básicas o básicas de la rama industrial como Fundamentos de Informática (FI), Fundamentos de Automática (FA) y Sistemas de Producción y Fabricación (SPF). En FI se solía impartir C, pero a partir del año pasado se ofrece una variante imperativa del C++, que hace hincapié en el uso más que en el diseño de clases y objetos. Esto abre la puerta a trabajar a un nivel abstracción mayor haciendo uso de elementos como la biblioteca STL. Por otra parte, el concepto de clase y de programación orientado objeto será beneficioso en el estudio de otros lenguajes más abstractos como aquellos que utilizan el concepto de bloques funcionales. En FA se dan los elementos básicos de la teoría de control de sistemas continuos. Mientras que en SPF (compartida por varios departamentos) se ofrece una colección variopinta de conceptos de interés sobre la industria contemporánea pero las limitaciones de tiempo impiden un tratamiento a fondo de los mismos.

El GIEIA es el que más posibilidades ofrece de desplegar el temario discutido en la sección anterior. Como se explicó en una contribución previa [1] los temas de la tabla 1 se repartieron entre las asignaturas de Informática Industrial (II) propiamente dicha y Control y Comunicaciones Industriales (CCI), referimos al lector a dicho artículo para discusión detallada, pero resumiendo mucho, en II se hace énfasis en el tema 4 de programación concurrente en C++ v11 utilizando las posibilidades nativas que ahora se ofrecen para la programación de hilos [21], [2] y una versión reducida del estándar POSIX para concurrencia mediante procesos, aplicados siempre a sistemas discretos. En CCI, una vez que el alumno ha dado sistemas muestreados en la asignatura obligatoria Diseño de Sistemas de Control, se está condiciones de aplicar los conceptos contenidos en los temas 5 y 6, aplicados ya a sistemas continuos. El resto de los temas permean todo el enfoque adoptado. El lenguaje que se ha utilizado es el C++ v11, un lenguaje de propósito general, que brinda un compromiso razonable entre nivel de abstracción y capacidad de bajar a nivel de hardware para las aplicaciones empotradas. La posibilidad de trabajar a otros niveles de abstracción como los bloques funcionales utilizados en los PLCs que cumple la norma IEC 61131-3 [22] o para sistemas distribuidos: norma IEC-61499 [23], no es discutida en profundidad puesto que objeto de estudio de otras asignatura como Automatización Industrial. Los temas más especializados de robótica, mecatrónica y visión artificial tienen sus propias asignaturas dedicadas. Se estudia la posibilidad de impartir bases de datos de tiempo real y de series temporales históricas, o así como el estándar OPC (clásico/UA) [24] que metodológicamente pertenecería a la capa de aplicación de un protocolo de comunicación, pero

que resultaría muy útil para la obtención de datos reales o simulados en el laboratorio.

En otras titulaciones la situación es más constreñida en término de créditos lectivos. En el Grado en Ingeniería Eléctrica (GIE) la asignatura de Informática Industrial Aplicada precede a otra asignatura obligatoria (Regulación Automática de Sistemas Eléctricos) y una opcional (Sistemas de Control Industrial) que podrían resultar complementarias. En cualquier caso, la estrategia seguida y considerada idónea dadas las circunstancias es la de establecer el marco presente y futuro (tema 1) de la disciplina para hacer énfasis en los sistemas de control industrial tipo PLCs, DCSs y SCADA y la programación en los estándares pertinentes, por una parte y en las redes industriales, sus requerimientos a partir del estándar OSI o similares, profundizando especialmente en el protocolo OPC por las ventajas prácticas ya comentadas.

En el Grado en Ingeniería Química (IQ) la situación es especialmente desfavorable. Se cuenta con una asignatura opcional, impartida después de Control y Simulación de Procesos Químicos que de alguna manera refuerza, con ejemplos de esa especialidad concreta, lo dado en FA. La estrategia seguida en este escenario es la de comentar someramente todos los temas delineados y estudiar con más detalle los sistemas de supervisión tipo SCADA. El OPC de nuevo como tecnología que propicia la movilidad de la información entre los diferentes niveles de la pirámide CIM. En este caso concreto se discuten conceptos de herramientas de ayuda a la decisión, muy valiosos para la titulación, relacionadas con el control estadístico de procesos multi-variantes.

En el Grado en Tecnologías en Industriales (GITI) se discuten la esencia de los temas recogidos en la tabla 1 en el marco de la asignatura, también opcional, llamada Informática Industrial en la que se sigue la tónica general planteada y se hace énfasis sobre los sistemas de control distribuido basados en el estándar IEC 61499.

### 5.2.1 Laboratorios

Las prácticas de laboratorio, como se ha recalado, resultan básicas. Existe el interés consciente de utilizar producto y lenguajes basados en normas y de ser posible, soluciones que estén libremente disponibles. Esto desde luego no es posible, y discutiblemente, tampoco es conveniente en todos los casos. Con el C++ v11 y librerías basadas en POSIX como vehículo para la enseñanza del temario, no habría mayores problemas. Pero tampoco se tiene porqué renunciar, por ejemplo, a MATLAB y las herramientas de modelado continuo y discreto que se éste ofrece y que brindan además la facilidad de desplegar, a partir del modelo creado, verificado y validado, soluciones en plataformas empotradas.

Existen además *toolboxes* de libre adquisición, como TrueTime [25] útiles a la hora de mostrar las bondades de uno u otro algoritmos de planificación de tiempo real en presencia de comunicaciones con simulación de pérdida de datos.

Por otra, los alumnos aprecian en gran medida, el ser capaces de practicar sobre productos comerciales de importancia acreditada y de amplia utilización en la industria, como pueden ser los PLCs de Siemens y los DCSs de Emerson (Delta-V). Software comercial como el generador de SCADAS *Wonderware Intouch* (Foxboro) o el sistema supervisor PI de OSISOFT, ofrecen un valor añadido que es muy apreciado por el alumno. En nuestras instalaciones de laboratorio, a partir de la existencia de estos productos comerciales, se pueden reproducir prácticamente todos los niveles desde el MES hacia abajo con productos como los mencionados y aplicarlos a plantas reales de laboratorio o a una simulación no trivial a tiempo real de una fábrica de azúcar de remolacha.

También existe la posibilidad de ejercer el control por computador aplicado a sistemas reales para aplicaciones mecatrónicas o de visión artificial.

## CONCLUSIONES

La informática industrial ha pasado de ser una mera herramienta tecnológica, a estar en el centro de la industria contemporánea y del previsible desarrollo futuro de la misma, concitando el interés no sólo de los ingenieros de la automatización sino también de la academia. La amplitud del tema, entre otras dificultades ya descritas, unidas a la escasa presencia del mismo en muchas titulaciones de ingeniería hace especialmente difícil la enseñanza de esta disciplina. En esta contribución se ha brindado un recorrido por los principales temas a tratar y se han discutido algunas ideas sobre el abordaje docente de los mismos en dependencia del tiempo disponible y de los objetivos y el contexto de la titulación.

### Agradecimientos

Este proyecto ha sido desarrollado en el marco de Proyecto de Innovación Docente (PID1617\_105) de la UVA titulado: La Informática Industrial en los Estudios de Ingeniería Industrial. También los autores desean agradecer el apoyo del proyecto del Ministerio de Educación titulado INOPTCON (MINECO/FEDER DPI2015-70975-P).

### Referencias

- [1] R. Mazaeda, E. de la Fuente, J. L. González, and E. J. Moya, "Sobre la docencia de la informática industrial," in *Actas de las XXXVI Jornadas de Automática*, Bilbao, 2015, pp. 688–695.
- [2] R. Mazaeda, E. de la Fuente, J. L. González,



- and E. J. Moya, “Docencia en Informática Industrial: Lenguajes de Programación,” in *Actas de las XXXVII Jornadas de Automática*, Madrid: Comité Español de Automática (CEA-IFAC), 2016, pp. 631–638.
- [3] F. Rodríguez, J. C. Moreno, M. Castilla, M. Berenguel, J. L. Guzmán, and J. A. Sánchez, “Experiencia docente de la materia Informática Industrial en estudios de Ingeniería,” in *Actas de las XXXVII Jornadas de Automática*, Comité Español de Automática (CEA-IFAC), 2016, pp. 762–768.
- [4] B. Scholten, *The Road to Integration: A Guide to Applying the ISA-95 Standard in Manufacturing*. ISA, 2007.
- [5] J. E. Hopcroft, R. Motwani, and J. D. Ullman, *Introduction to Automata Theory, Languages, and Computation (3rd Edition)*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2006.
- [6] Wirth, “Algorithms and Data Structures,” *Princet. Univ. COS*, vol. 1985, no. August, pp. 5–183, 2007.
- [7] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-oriented Software*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1995.
- [8] K. J. Åström and P. R. Kumar, “Control: A perspective,” *Automatica*, vol. 50, no. 1, pp. 3–43, 2014.
- [9] B. Wittenmark, K. J. Åström, and K. E. Årzen, “Computer Control: An Overview,” *IFAC Prof. Br.*, pp. 1–82, 2002.
- [10] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu, “A Survey of Recent Results in Networked Control Systems,” *Proc. IEEE*, vol. 95, no. 1, pp. 138–162, 2007.
- [11] P. J. Ramadge and W. M. Wonham, “The control of discrete event systems,” *Proc. IEEE*, vol. 77, no. 1, pp. 81–98, 1989.
- [12] C. Seatzu, M. Silva, and J. H. van Schuppen, *Control of Discrete-Event Systems*, vol. 433. Springer, 2013.
- [13] E. Ashford Lee and P. Varaiya, \**Structure and Implementation of Signals and Systems*. 2011.
- [14] K.-D. Kim and P. R. Kumar, “An Overview and Some Challenges in Cyber-Physical Systems,” *J. Indian Inst. Sci.*, vol. 93, no. 3, pp. 341–352, 2013.
- [15] Y. Lu, “Industry 4.0: A Survey on Technologies, Applications and Open Research Issues,” *J. Ind. Inf. Integr.*, 2017.
- [16] A. J. C. Trappey, C. V. Trappey, U. Hareesh Govindarajan, A. C. Chuang, and J. J. Sun, “A review of essential standards and patent landscapes for the Internet of Things: A key enabler for Industry 4.0,” *Adv. Eng. Informatics*, 2016.
- [17] P. Neumann, “Communication in industrial automation—What is going on?,” *Control Eng. Pract.*, vol. 15, no. 11, pp. 1332–1347, 2007.
- [18] S. Dormido, J. Sánchez, and E. Kofman, “Muestreo, Control y Comunicación Basados en Eventos,” *Rev. Iberoam. Automática e Informática Ind. RIAI*, vol. 5, no. 1, pp. 5–26, 2008.
- [19] M. Silva, “Half a century after Carl Adam Petri’s Ph.D. thesis: A perspective on the field,” *Annu. Rev. Control*, vol. 37, no. 2, pp. 191–219, 2013.
- [20] R. Goebel, R. G. Sanfelice, and a. Teel, “Hybrid dynamical systems,” *IEEE Control Syst.*, vol. 29, no. 2, pp. 28–93, 2009.
- [21] A. Williams, *C++ Concurrency in Action. Practical Multithreading*. Manning, 2012.
- [22] A. Otto and K. Hellmann, “IEC 61131: A general overview and emerging trends,” *IEEE Ind. Electron. Mag.*, vol. 3, no. 4, pp. 27–31, 2009.
- [23] V. N. Dubinin and V. Vyatkin, “Semantics-robust design patterns for IEC 61499,” *IEEE Trans. Ind. Informatics*, vol. 8, no. 2, pp. 279–290, 2012.
- [24] M. H. Schwarz and J. Borcsok, “A survey on OPC and OPC-UA: About the standard, developments and investigations,” *2013 XXIV Int. Conf. Information, Commun. Autom. Technol.*, pp. 1–6, 2013.
- [25] A. Cervin, D. Henriksson, B. Lincoln, J. Eker, and K.-E. Arzen, “How does control timing affect performance? Analysis and simulation of timing using Jitterbug and TrueTime,” *IEEE Control Syst.*, vol. 23, no. 3, pp. 16–30, Jun. 2003.