

# Capítulo 9

## Industria 4.0 en Robótica

Irene Fassi, Claudia Pagano y Serena Ruggeri

### 9.1 Implantación

La llamada 4ª revolución industrial va a cambiar radicalmente la idea común de la fábrica, y un gran esfuerzo de investigación industrial está dedicado a la definición de los modelos de “Fábrica del Futuro”.

La cuarta revolución industrial se caracteriza por la palabra *automatización*, que se logra en la producción industrial al abordar cuatro desafíos:

1. Las máquinas inteligentes controlan la producción en fábricas inteligentes en las que la presencia de humanos se reduce enormemente.
2. Las máquinas inteligentes optimizan la capacidad de la instalación de producción, coordinando el movimiento del material, analizando el estado de la cadena de producción y el stock, y reduciendo a cero el tiempo de inactividad.
3. Las máquinas inteligentes son esencialmente autoorganizadas; la planificación de materiales y el manejo de pedidos están completamente automatizados
4. Las máquinas inteligentes pueden reconfigurar de forma autónoma la línea de producción para responder en muy poco tiempo a la solicitud personalizada del cliente.

Las máquinas inteligentes se basan en una parte cibernética compleja y un algoritmo de inteligencia artificial. De hecho, Industria 4.0 tiene principalmente dos fundamentos: la capa de hardware, que incluye todos los elementos físicos, y una capa de software, la parte cibernética. En el capítulo anterior se discutieron la

---

I. Fassi, C. Pagano(✉), S. Ruggeri

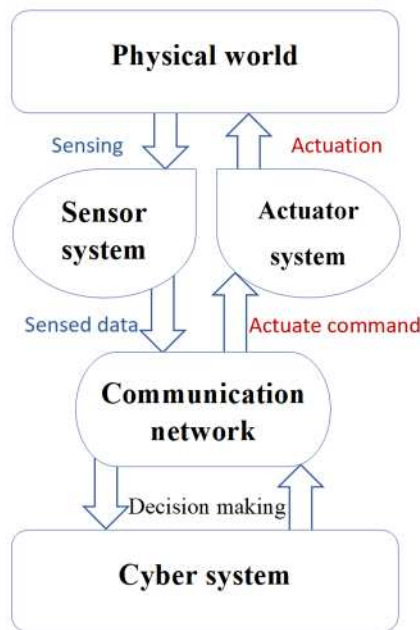
Instituto de Tecnologías y Sistemas Industriales Inteligentes para la Fabricación Avanzada,  
Consejo Nacional de Investigación, Milán, Italia  
correo electrónico: claudia.pagano@stiima.cnr.it

© Los Autores 2020

J.C. Chaplin et al. (eds), *Fabricación digital para pymes*

detección de robots y la actuación, que pertenecen a la capa de hardware. En este capítulo se discute la parte cibernética, centrándose en la programación, el control y la inteligencia artificial. Además, al final del capítulo se proporciona una breve descripción de los problemas éticos relacionados con la inteligencia implementada en las máquinas.

## 9.2 Sistema ciber-físico



**Figura 9.2-1** Vista holística de CPS

Las fábricas inteligentes requieren el despliegue sinérgico de varias tecnologías, y entre ellas el *Sistema Físico Cibernético* (CPS) es esencial.

El elemento clave para CPS es la *interacción* entre elementos físicos (mundo real) y cibernéticos (mundo virtual), que son sistemas cooperadores, que tienen comportamientos autónomos, conciencia del contexto a través de capacidades de detección mejoradas y capacidades de almacenamiento y procesamiento de datos de los sensores de la red (Figure 9.2-1).

Algunos de los ejemplos prácticos que ya han surgido incluyen la robótica avanzada (por ejemplo, automóviles autónomos, aviónica piloto automática, microrrobótica, cirugía asistida por robot, dispositivos médicos implantados), edificios inteligentes y la red eléctrica inteligente.

Los sistemas ciber-físicos (CPS) crean un vínculo entre los *sistemas físicos y digitales* con el fin de generar una infraestructura común con capacidad avanzada. Permiten la integración de la dinámica de los procesos físicos con los del software y las redes, para que puedan ser manejados como una sola entidad.

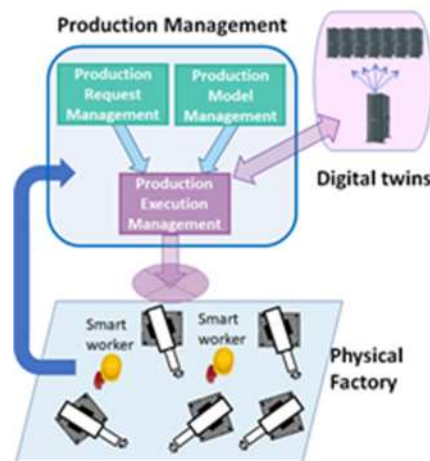
Una CPS realiza dos tareas principales:

- *Conexión intensa*: Con el fin de garantizar la adquisición continua de datos del mundo físico y la retroalimentación de la información del espacio cibernético.
- Gestión de datos, análisis y computación: Para crear el espacio cibernético.

Para realizar estas tareas, un CPS está compuesto por "entidades computacionales colaboradoras que conectan el mundo cibernético con los entornos o procesos físicos circundantes en un entorno de Internet" [1]. Una CPS incluye "sistemas integrados (como equipos, edificios, medios de transporte y dispositivos médicos), servicios de Internet, logística, coordinación y procesos de gestión" [1]. Una amplia gama de sensores y actuadores se utilizan para conectar los elementos de la CPS y permitir la comunicación hombre-máquina. La CPS almacena y procesa todos los datos recibidos de los sensores y sistemas de comunicación, y controla los sistemas físicos utilizando los actuadores.

La investigación y el desarrollo en esta área están abordando una amplia gama de cuestiones, incluida la inferencia a partir de datos empíricos (es decir, datos empíricos elaborados para sacar conclusiones), la detección y la percepción, el aprendizaje motor y el control para adaptarse a diferentes contextos, y el diseño, implementación y verificación de CPS seguros y que funcionen correctamente.

Un esquema simplificado de un CPS y la interacción entre humanos y máquinas se muestra en Figure 9.2-2.



**Figura 9.2-2** Esquema CPS con conexiones de interacción.

Los *sistemas de producción física cibernética* (CPPS) son sistemas físicos cibernéticos para la producción. Permiten y apoyan la comunicación entre humanos, máquinas y productos. El CPPS consiste en elementos autónomos y cooperativos -relacionados con procesos, máquinas, producción y logística- conectados a todos los niveles de producción, y un sistema de información para que sus actividades de operación y cooperación puedan modelarse. Gracias al intercambio continuo de datos entre y dentro de los sistemas, almacenados y analizados en tiempo real, el modelo siempre está sincronizado con el estado de la fábrica y el comportamiento de todo el sistema se puede pronosticar en función de situaciones pasadas y presentes. Esto puede ayudar en varios procesos de toma de decisiones para que las acciones más adecuadas se puedan implementar rápidamente, mejorando la productividad de la fábrica inteligente [2].

Las principales características de CPPS son:

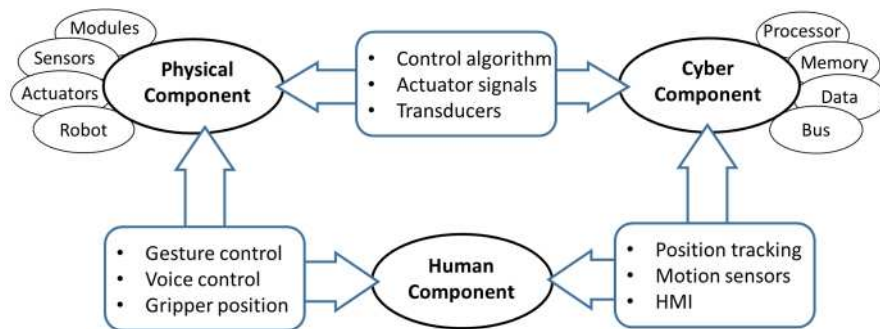
- *Inteligencia*: Los componentes del sistema son capaces de adquirir información del entorno y actuar de forma autónoma.
- *Conectividad*: Los elementos del sistema son capaces de cooperar y colaborar entre sí y están conectados a los conocimientos y servicios en Internet.
- *Responsabilidad*: El sistema es capaz de reaccionar a los cambios internos y externos.

El concepto CPPS está fuertemente vinculado a otros conceptos innovadores como Internet de las cosas (IoT), *big data* y gemelos digitales. El *IoT* se refiere a una capacidad de comunicación inalámbrica integrada con sensores y computación que permite la recopilación de datos relacionados con objetos identificables de forma única a través de Internet. *Big data* se refiere a un nuevo paradigma informático que permite la recopilación, procesamiento y análisis de cantidades masivas de datos [3].

El Digital Twin (DT) es otra tecnología estrictamente relacionada con la Industria 4.0 y la fabricación inteligente. Al igual que con CPS, el gemelo digital está asociado con la integración entre los mundos cibernético y físico. Por definición, un IME crea un modelo virtual de un sistema físico para predecir el comportamiento real basado en la simulación en tiempo real del sistema. Por lo tanto, los dos conceptos tienen objetivos y enfoques similares. Sin embargo, mientras que el CPS se implementa con sensores y actuadores, que permiten la interacción entre el mundo físico y cibernético, DT se basa en modelos y datos. Los gemelos digitales son réplicas digitales de sistemas físicos; representan perfiles digitales en evolución que están diseñados utilizando los datos recopilados del comportamiento pasado y presente de un objeto o proceso físico. Se aplican principalmente en tareas como el monitoreo, el mantenimiento predictivo y la optimización de sus contrapartes físicas.

### 9.2.1 CPS colaborativo

Los CPS colaborativos (CCPS) son sistemas de producción aún más complejos con respecto a otros CPS: el concepto de “integralidad” representa el nivel de integración de los diferentes módulos, con el objetivo de propiedades de autoaprendizaje y reconfiguración. Para mejorar esta integración, que concierne también a la Interfaz Hombre-Máquina (HMI), el CCPS debe presentar un alto nivel de “sociabilidad”, es decir, las capacidades de comunicación con el entorno, incluidas otras CPS [4].



**Figura 9.2-3** Estructura de un Sistema Ciber-físico Robótico Colaborativo.

CCPS incluye tres entidades principales: el componente humano (HC), el componente físico (PC) y el componente computacional (CC), que están interconectados a través de una variedad de tecnologías (Figure 9.2-3). Al ser parte de un sistema ciber-físico, un humano y un robot pueden interactuar y realizar tareas de autoorganización. En un escenario futuro, la red de sensores integrada y las tecnologías de comunicación de CPS hacen que la interacción sea confiable, segura y protegida. Varios sensores permiten que el robot sea consciente de la presencia humana y reaccione consecuentemente, basándose en el esquema de evitación humana implementado. Para que la interacción sea segura, es esencial un sistema de seguimiento de posición humana en tiempo real y preciso. Un sistema de visión conectado con el robot y monitoreando su entorno puede proporcionar información sobre la ubicación del trabajador, que se utiliza como entrada para los esquemas de reducción de velocidad para el robot. Al mismo tiempo, el sistema de monitoreo integrado con un sistema de reconocimiento de gestos puede ser utilizado por el trabajador para controlar el robot, junto con un control por voz. Además, se pueden integrar sensores de fuerza en el sistema para mejorar el nivel de interacción del robot humano. Combinando los datos de los sensores de fuerza y el sistema de visión, la velocidad y la aceleración del robot se pueden modular de acuerdo con la parte del cuerpo del trabajador en las proximidades del efector final. Los sensores de fuerza pueden proporcionar una característica adicional que permite un contacto entre el humano y el robot para que el trabajador entrene al robot con la mano.

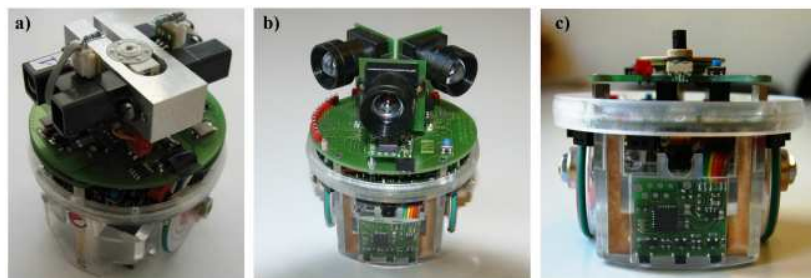
### 9.2.2 Robótica de enjambre y CPS

Una aplicación temprana del concepto de CPS se puede encontrar en la robótica enjambre, un tema ampliamente estudiado en los últimos 20 años. La robótica de enjambres está inspirada en la naturaleza y particularmente en el comportamiento de enjambres capaces de realizar tareas que estarían más allá de las capacidades de los individuos. Se trata de la coordinación de equipos robóticos distribuidos, interconectados, que pueden considerarse, por lo tanto, CPS simplificados.

Se han realizado varios trabajos [5], estudiando y mejorando la comunicación entre robots y el algoritmo de gestión de la estrategia de grupo. Por razones prácticas (el espacio disponible), analogía con la naturaleza (enjambre de insectos), pero también razones técnicas, este enfoque se estudia ampliamente en un gran número de mini y micro robots; de hecho, debido a la limitación del consumo de espacio y energía, los robots miniaturizados típicamente tienen un grado de complejidad reducido. Por lo tanto, la estrategia grupal es muy adecuada para robots tan pequeños y simples.

Un ejemplo muy exitoso de robótica de enjambre se ha desarrollado alrededor del e-puck, un minirobot móvil desarrollado principalmente con fines educativos [6], y una variedad de trabajos para estudiar la coordinación y sincronización de grupos [7][8][9].

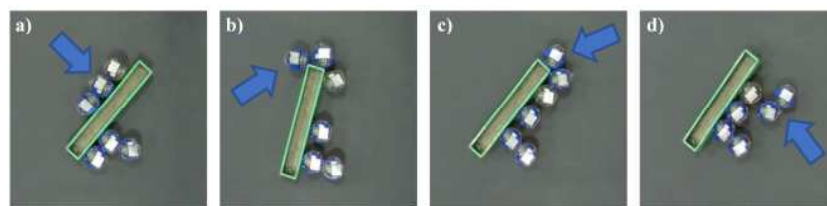
El e-puck [10] tiene una sección circular de 75 mm de diámetro y altura variable, dependiendo de las extensiones proporcionadas (Figure 9.2-4).



**Figura 9.2-4** Configuración del robot E-puck provista de: a) un escáner de distancia IR, b) cámara lineal de gran campo de visión y c) sensor de medición de color de tierra [6].

Entre estos trabajos algunos se han centrado en la estrategia para hacer un número adecuado de e-puck capaz de mover una caja y colocarla en una posición elegida. Aunque el diseño del robot permite su personalización proporcionándole los sensores y características más adecuados requeridos para la tarea (Figure 9.2-5), uno de los objetivos de las obras es probar varios algoritmos, no solo para lograr la tarea, sino también para hacerlo con la configuración más fácil de los robots como sea posible.

Los objetos a transportar fueron elegidos para que fueran lo suficientemente pesados como para requerir los esfuerzos de todo el grupo, por lo que la cooperación era esencial. Los robots están equipados con varios sensores IR para detectar los obstáculos y encontrar la posición objetivo, una cámara para identificar los obstáculos (los objetos y los otros robots) y una cámara óptica con baja resolución de imagen debajo del robot. Esta cámara da a los robots individuales la percepción de la dirección del movimiento del objeto, de acuerdo con esto los robots cambian el punto de aplicación de su empuje, evitando neutralizar la fuerza de otro robot. Eventualmente, todos están alineados para transportar con éxito el objeto [7].



**Figura 9.2-5** Alineación de los robots e-puck para transportar con éxito el objeto [11].

Una aplicación más práctica de los algoritmos típicos de la robótica de enjambre, y específicamente la planificación de rutas y la coordinación de equipos robóticos distribuidos en un CPS, está representada por almacenes automatizados. Como ejemplo, en los almacenes de Amazon y Ocado, todo el sistema, desde el pedido hasta la entrega, está automatizado. Un gran número de robots móviles autónomos (Figure 9.2.6) están conectados entre sí y con el sistema donde se colocan los pedidos y un algoritmo optimiza la trayectoria de los robots de acuerdo con las posiciones de los bienes pedidos para que el desplazamiento de los robots esté programado para mover los bienes y empacar el pedido. La conexión y coordinación entre los robots es esencial para evitar posibles accidentes.

En el almacén de Amazon [12], los robots mueven los estantes que contienen los bienes necesarios hacia un trabajador, que recoge los bienes correctos sin el agotador paseo entre los estantes.

En el almacén de Ocado [13], los robots colocan cajas y recogen productos de ellos moviéndose a lo largo de un sistema de cuadrícula.

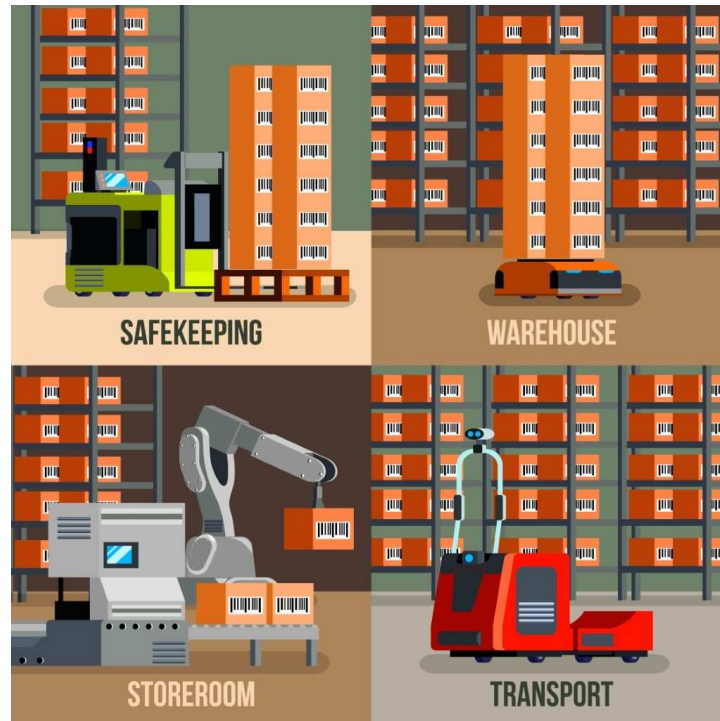


Figura 9.2-6 Almacén automatizado.

### 9.3 Programación del robot

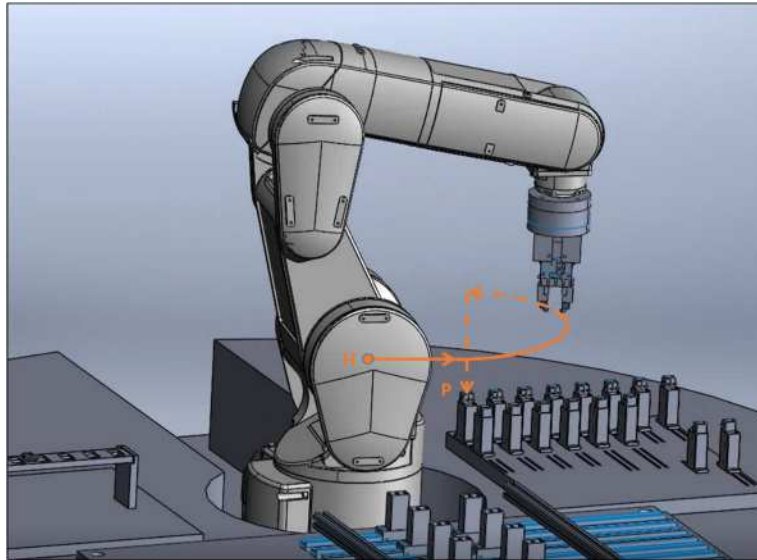
La ejecución de una tarea completa en una aplicación robótica requiere la ejecución de una o más operaciones por parte del robot. El robot tiene que moverse de acuerdo con un movimiento específico que está definido total o parcialmente por la ley de movimiento impuesta a su efector final. El movimiento deseado es el resultado de una *planificación de movimiento adecuada* y la *programación de la tarea del robot* y se logra mediante un conjunto de comandos enviados a los actuadores del robot de acuerdo con una estrategia de control predefinida. Para la correcta ejecución de la tarea, es importante conocer la estructura cinemática del robot, incluyendo el tipo y la ubicación de los actuadores (motores), las juntas, las transmisiones y las dimensiones de la estructura mecánica. Tal conocimiento se completa con el de la electrónica que controla los actuadores y de los límites eléctricos y mecánicos de los componentes.



### 9.3.1 Planificación de movimiento

La posición y el movimiento del efector final o la fuerza que ejerce depende de la posición, el movimiento y las acciones que las diferentes partes del robot ejercen por medio de sus actuadores. Por lo tanto, el análisis del robot y su modelización son pasos preliminares en el desarrollo de una aplicación robótica.

El problema con la planificación de una trayectoria para un robot se puede dividir en encontrar un *camino* y definir una *ley de temporización* en el camino. De hecho, la ruta identifica el lugar de los puntos que el robot debe seguir para realizar el movimiento deseado (Figure 9.3-1), mientras que la "*trayectoria es un camino en el que se especifica una ley de temporización*" [14], por ejemplo, en términos de velocidades en cada punto.



**Figura 9.3-1** Ejemplo de un robot en movimiento simulado, moviéndose desde un punto inicial (H – posición de inicio) hacia el punto P para recoger un componente sobre la mesa. Derechos de la imagen del autor.

Por lo tanto, la *planificación del movimiento* se refiere a la generación de las leyes de temporización para las coordenadas (en las articulaciones o en el efector final), por lo tanto, las entradas al sistema de control de movimiento [14].

Dependiendo de las operaciones que el robot deba realizar, el robot puede adoptar una de las siguientes estrategias de movimiento[15]: desde un punto inicial hasta un punto final (*movimiento punto a punto*) con una trayectoria arbitraria; a través de una secuencia finita de puntos asignados a lo largo del camino (*movimiento a través de puntos intermedios*); o de acuerdo con una *trayectoria específica*.

Al definir la ley de temporización, se deben considerar varios aspectos, que incluyen [14] [15]:

- Las *restricciones* derivadas del tipo de aplicación.
- La estructura mecánica del robot.
- Las *fuerzas conjuntas y los pares de torsión* deben respetar los límites de los actuadores.
- Las unidades y el controlador.
- Las *trayectorias* que interpolen los puntos de trayectoria deben ser *suaves* para minimizar los efectos no deseados, como las vibraciones, las trayectorias no deben implicar una *carga computacional* muy alta.
- Las *posiciones y velocidades en las articulaciones* deben ser funciones continuas del tiempo (la continuidad de las aceleraciones no es obligatoria, aunque generalmente bienvenida).

### 9.3.2 Programación

La *programación* de un robot significa instruirle sobre la tarea (s) que tiene que realizar. Se debe definir un entorno de programación y un lenguaje o lenguajes de programación adecuados.

Los *métodos de programación* existentes incluyen [16][17][18] diferentes enfoques:

- *Guiando el robot a las posiciones de interés* o a lo largo de los caminos deseados (programación de enseñanza) utilizando un colgante de enseñanza, sosteniendo manualmente el efector final, o en teleoperación (utilizando un sistema maestro-esclavo, de modo que el operador mueva el robot maestro y el robot "esclavo" real se mueve en consecuencia).
- *Escribir un programa* en un lenguaje que el robot pueda interpretar y ejecutar (lenguaje propietario o bibliotecas robóticas que admitan lenguajes de programación estándar).
- *Utilizando interfaces gráficas interactivas*, es decir, modelado CAD (*Computer Aided Design*) y herramientas de simulación de sistemas para definir la tarea y generar el programa automáticamente.
- La *programación a nivel de tareas* consiste en que el usuario le dice al robot lo que debe hacerse y el robot sabe cómo hacerlo. Esto se logra un modelado y detección exhaustivos del medio ambiente y la inteligencia de la máquina. Por lo tanto, la programación a nivel de tareas es difícil de lograr en la actualidad, pero posiblemente se desarrollen futuras aplicaciones innovadoras.

La generación de programas de robots a partir de paquetes de software CAD / CAM se utiliza con frecuencia, ya que son muy comunes entre las empresas de fabricación [16]. En el contexto de la fábrica digital, las soluciones digitales para la representación y simulación 3D de un robot individual o líneas robóticas apoyan

efectivamente el diseño y la validación del proceso de fabricación. Otros enfoques para instruir a un robot incluyen, por ejemplo, usar dispositivos señaladores o coreografiar los movimientos de la tarea [16][18].

Los métodos de programación de robots se pueden separar en *programación en línea* y *programación fuera de línea* (off-line).

Con la *programación en línea*, se utiliza el robot físico y el operador actúa directamente sobre el controlador del robot. La programación por enseñanza pertenece a esta categoría. El robot se enseña o se guía según sea necesario y las acciones se registran en la memoria del controlador del robot. El robot entonces ejecuta los movimientos de una manera repetitiva.

La programación en línea es bastante fácil y no requiere habilidades de operador especializado. No es sensible a los errores de precisión. Sin embargo, requiere acceso al robot, lo que significa que la producción se cierra durante la fase de programación. Además, la edición del programa (por ejemplo, para agregar algunos parámetros, corregir una línea o agregar interacciones con otros dispositivos) puede ser limitada, lo que significa que las tareas complejas pueden ser difíciles de configurar. Finalmente, los programas del robot solo se almacenan en la memoria del robot, por lo que puede ser de difícil acceso. Este método puede ser útil si se requiere que el robot ejecute repetidamente una tarea durante un largo período de producción. La programación por idioma en línea también es posible: en este caso el operador escribe los comandos directamente en el colgante de enseñanza, pero puede ser difícil realizar en ese momento todos los efectos, como posibles colisiones.

En la *programación fuera de línea*, se utilizan herramientas de software (programación por lenguaje o interfaces gráficas interactivas) para generar el programa (sin la participación física del robot) que luego se implementa en el controlador del robot. Con la programación fuera de línea, las posibilidades de edición aumentan, el robot real puede funcionar durante la fase de programación para que la producción no se cierre, y la puesta en marcha y el cambio de producto son más rápidos. Sin embargo, el robot debe ser preciso (necesidad de una calibración del robot) y se requiere formación del operador especializado.

Una combinación de programación en línea y fuera de línea también es posible.

### 9.3.3 Lenguajes y entornos de programación

Hay muchos *lenguajes de programación* utilizados en robótica que se han adoptado a lo largo de los años. Algunos ejemplos de los primeros lenguajes de programación utilizados fueron BASIC, Pascal y LISP. Si bien estos lenguajes ahora están obsoletos, proporcionaron la base para lenguajes robóticos industriales más recientes. Otros lenguajes de scripting comunes incluyen C, C ++, C #, .NET, Python, Matlab y Java, mientras que un ejemplo de un lenguaje de programación visual es Labview.

Además, existen muchos *lenguajes propietarios* desarrollados por diferentes fabricantes e incluso para robots similares pueden ser muy diferentes. Actualmente,

casi todos los fabricantes de robots tienen su propio lenguaje de programación propietario. Algunos ejemplos son: RAPID por ABB, MELFA-BASIC por Mitsubishi, KRL por Kuka, AS por Kawasaki, VAL3 por Stäubli, PDL2 por Comau, Karel por Fanuc, Inform por Yaskawa y URScript por Universal Robots.

Los fabricantes también proporcionan *entornos de programación para la simulación y la programación fuera de línea*. El robot y toda la celda de trabajo se pueden representar como un modelo CAD 3D. Los movimientos del robot, la colisión del brazo y las acciones de la herramienta se pueden simular y la tarea definida y optimizada. Algunos ejemplos de entornos de programación de los fabricantes son RobotStudio [19] de ABB y RT Toolbox3 Pro [20] de Mitsubishi.

Sería claramente deseable “*desarrollar bibliotecas robóticas que se utilicen en el contexto de normas consolidadas o nuevos lenguajes de propósito general para aplicaciones de automatización industrial*” [17]. De hecho, una de las actividades actuales de investigación y desarrollo destinadas a apoyar la implementación de la Industria 4.0 es el desarrollo o uso de estándares relacionados con herramientas de programación, modelización y simulación, protocolos de comunicación e interfaces.

En los últimos años, se ha desarrollado un software de programación fuera de línea de terceros que admite todas las principales marcas robóticas para proporcionar un marco común útil para la simulación y programación de diferentes robots en muchas aplicaciones con una sola herramienta. Algunos ejemplos incluyen RoboDK [21], Robotmaster [22], Delfoi Robotics [23], FASTSUITE [24] y OCTOPUZ [25].

Además de los productos comerciales, existen algunas soluciones de código abierto. Entre estos, *ROS* (Robot Operating System) se está volviendo muy común. ROS proporciona un marco común para aplicaciones de robótica. Es un sistema meta-operativo que proporciona los servicios esperados de un sistema operativo, “*incluyendo la abstracción de hardware, control de dispositivos de bajo nivel, implementación de funcionalidades comúnmente utilizadas, paso de mensajes entre procesos y administración de paquetes. También proporciona paquetes de herramientas y bibliotecas para obtener, crear, escribir y ejecutar código en varios equipos*” [26].”

“*ROS-Industrial es un módulo ROS de código abierto que extiende las capacidades avanzadas de ROS a la automatización de la fabricación y la robótica. El repositorio ROS-Industrial incluye interfaces para robots industriales comunes, pinzas, sensores y redes de dispositivos*” [27].” Proporciona bibliotecas de software para, por ejemplo, calibración automática de sensores 2D / 3D y planificación de movimiento.

## 9.4 Control del robot

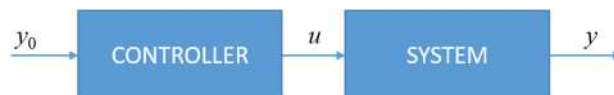
A nivel de robot, es necesario implementar una arquitectura de control y supervisión, capaz de interactuar con el entorno externo dentro del CPPS y

transformar los estímulos externos en comandos de actuadores. Además, se necesita un dispositivo para regular las acciones generadas por los actuadores para que el robot se comporte como se desea: este dispositivo se llama *controlador*. Recibe el comportamiento del robot deseado como entrada  $y$ , para determinar las acciones de los actuadores, a menudo requiere información sobre el estado del robot (posición, velocidad, fuerza intercambiada, etc.) que se obtiene mediante la lectura de los datos proporcionados por los sensores adecuados. Cuando esta función de monitoreo está presente, este tipo de control se denomina *control de bucle cerrado*, de lo contrario se denomina *control de bucle abierto*.

El sistema de control es necesario para determinar el historial de tiempo de las fuerzas o pares a desarrollar por los actuadores conjuntos para garantizar la ejecución de la tarea comandada [28]. En términos simples, por ejemplo, si  $y$  es la posición real del robot,  $y_0$  la posición deseada del robot, y  $u$  la fuerza o el par generado por el motor del robot: el controlador tiene que determinar  $u$  para hacer que el robot se mueva de modo que  $y$  coincida con  $y_0$ .

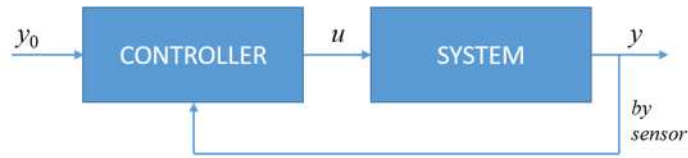
Para comprender mejor el tema, primero se pueden identificar los elementos principales de un esquema de control: el "sistema" a controlar y el "controlador" que proporciona una acción de control al sistema. El sistema a controlar se puede ver como una caja que responde a una entrada  $u(t)$  produce una salida  $y(t)$ . El objetivo es hacer que el sistema produzca una salida como la entrada del controlador deseada  $y_0(t)$  (por ejemplo, para hacer que un eje de un robot se mueva de la manera deseada).

Existe una distinción importante en la teoría del control entre el control en bucle abierto y el control en bucle cerrado, es decir, el control con un bucle de retroalimentación. En el control de bucle abierto (Figure 9.4-1), el controlador calcula la acción de control adecuada  $u(t)$  para producir una salida  $y(t)$  igual a  $y_0(t)$  sobre la base de la relación conocida entre la entrada y la salida del sistema.



**Figura 9.4-1** Esquema general de control de bucle abierto.

Sin embargo, a menudo falta un conocimiento adecuado del sistema, y las perturbaciones impredecibles o parcialmente conocidas (por ejemplo, efectos de acoplamiento entre las articulaciones) pueden actuar sobre el sistema. Por lo tanto, un enfoque común consiste en cerrar el bucle de control con la información de retroalimentación sobre el valor real de la salida procedente de los sensores (Figure 9.4-2). De esta manera, el controlador calcula la entrada al sistema para reducir al máximo el error entre las salidas deseadas y reales (p. ej., posición deseada y posición real del motor).

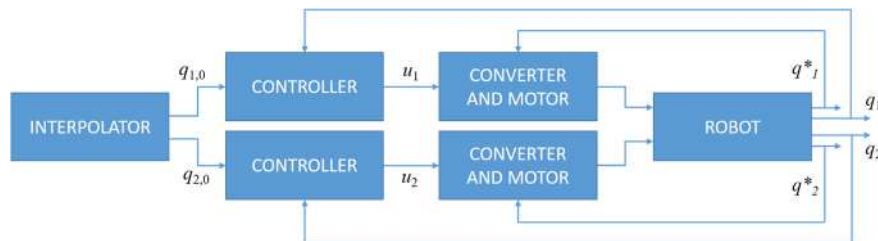


**Figura 9.4-2** Esquema general de control de circuito cerrado.

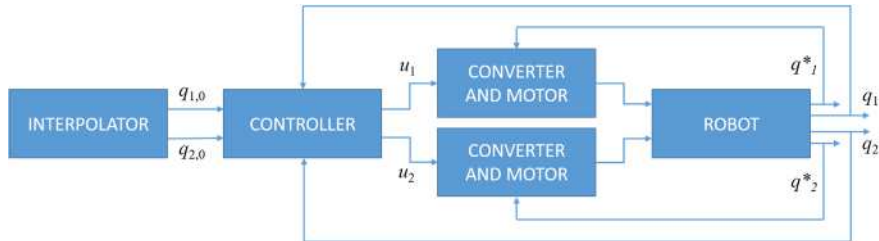
En otras palabras, la salida deseada se compara con la variable de retroalimentación proporcionada por un sensor adecuado (por ejemplo, codificador), luego el controlador define la acción sobre la base del error entre ellos. El cierre del lazo permite una mejor acción, tal como robustez y reducción de perturbaciones.

Mover el robot requiere el control de diferentes motores (un robot generalmente tiene más de un grado de libertad), por lo tanto, se necesitan métodos de control de varios ejes. Dependiendo de la forma en que se consideren y controlen los diferentes ejes, las estrategias de control se pueden dividir en [28][29]:

5. *Control descentralizado*: Cada articulación se considera independiente de otras articulaciones y controlada por un bucle de control independiente (Figure 9.4-3).
6. *Control centralizado*: En este caso el robot se considera como un sistema multivariable. El modelo dinámico del robot a menudo se tiene en cuenta (Figure 9.4-4).



**Figura 9.4-3** Esquema de control descentralizado: ejemplo con 2 ejes, donde  $q_i$  ( $i = 1, 2$ ) representa la posición de la junta y  $q^*_i$  la velocidad de la junta. El bloque interpolador calcula los movimientos deseados requeridos para cada eje.



**Figura 9.4-4** Esquema de control centralizado: ejemplo con 2 ejes, donde  $q_i$  ( $i = 1, 2$ ) representa la posición de la junta y  $q^*_i$  la velocidad de la junta. El bloque interpolador calcula los movimientos deseados requeridos para cada eje.

En el primer caso, las acciones de cada actuador solo dependen de los movimientos reales y deseados de *ese* actuador, mientras que en el caso del control centralizado las acciones de cada actuador dependen de los movimientos deseados y reales de *todos* los actuadores.

Esto también distingue la forma en que se consideran los efectos de interacción dinámica y acoplamiento entre las articulaciones. En el primer caso, las interacciones dinámicas se tratan como perturbaciones, mientras que en el segundo los efectos de acoplamiento pueden tenerse realmente en cuenta.

Los métodos de control de varios ejes a menudo descansan en controladores de un solo eje que controlan un motor cada uno, lo que permite un buen rendimiento en muchas aplicaciones. Las estrategias de control centralizado generalmente se implementan cuando los efectos mutuos entre las articulaciones son significativos y deben considerarse.

Entre los controladores centralizados, se pueden considerar diferentes enfoques, como el *control de par calculado* y el *control de dinámica inversa*. Estos controladores se basan en el modelo dinámico del robot. Si este modelo no es bien conocido, el rendimiento de estos esquemas de control apenas puede mejorar. Por esta razón, una buena estimación de los valores de los parámetros dinámicos del sistema (por ejemplo, masas, inercias, etc.) es necesario. Tal estimación se puede lograr mediante la calibración dinámica del robot.

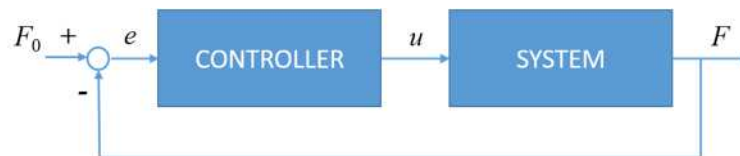
Además, es importante tener en cuenta que los robots reales difieren de los ideales debido a la presencia de diferentes fenómenos, incluida la inercia variable, la fricción estática y el cumplimiento que deben tenerse en cuenta al diseñar el esquema de control.

#### 9.4.1 Control basado en la fuerza y la visión

Hay casos en los que se requiere un robot para intercambiar fuerzas con el medio ambiente, por ejemplo, para presionar una pieza de trabajo, para insertar un pasador en un orificio o para desbarbar una pieza de trabajo. En estos casos, controlar la fuerza de contacto es más conveniente que controlar el movimiento. Otra rama del control robótico está constituida por el *control de fuerza*. [31][32]

El control de fuerza se puede dividir en:

- *Control directo*: La fuerza se controla cerrando el bucle de retroalimentación de la fuerza. El esquema general de un controlador de fuerza se ilustra en Figure 9.4-5.
- *Control indirecto*: El control de la fuerza se consigue mediante el control de movimiento (control de una deformación). La impedancia o el control de admisión pertenecen a esta categoría.



**Figura 9.4-5** Esquema general de control de fuerza directa.

En control directo, la retroalimentación de la fuerza es proporcionada por un transductor de fuerza en el efector final, en las articulaciones o en la punta de los dedos de la mano del robot cuando se requiere una fuerza adecuada para agarrar y manipular un objeto.

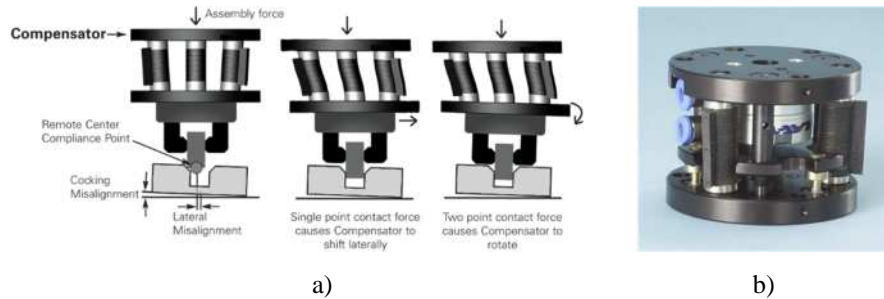
Es importante tener en cuenta que no es posible controlar tanto el movimiento como la fuerza en la misma dirección. Cuando la tarea de interacción identifica direcciones a lo largo de las cuales se debe controlar el movimiento y otras direcciones donde la fuerza es la variable controlada, se utilizan esquemas *híbridos de control de fuerza / movimiento* [32][33].

Debido a su complejidad, a veces los controladores de posición más simples se combinan con un cumplimiento pasivo en el efector final, de modo que los pequeños errores de posición pueden ser compensados por estos elementos elásticos. Un ejemplo es el Remote Center of Compliance (RCC), que es un dispositivo mecánico que comúnmente se monta entre la pinza y la muñeca del robot (Figure 9.4-6).

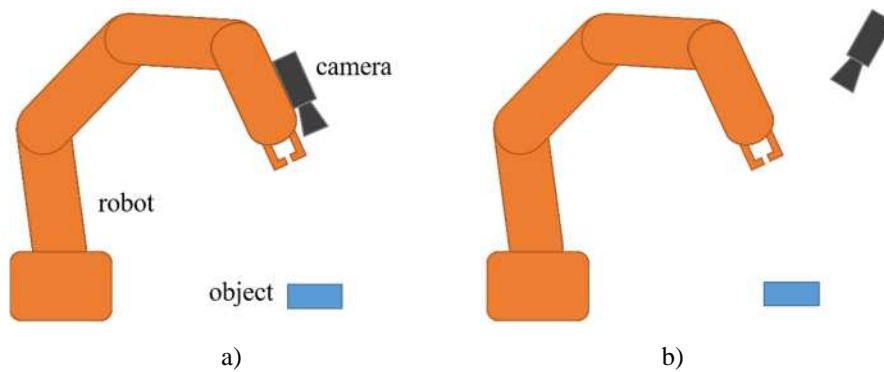
La retroalimentación de fuerza también es útil para la *teleoperación* del robot. En el modo teleoperado, el robot es mandado remotamente por el operador utilizando un enfoque maestro / esclavo. La idea es que el operador mueva un dispositivo (el maestro, por ejemplo, un joystick) y el robot (el esclavo) actúa de la misma manera. En este caso, los *interfaces hápticos* se pueden utilizar para proveer del operador la reacción de la fuerza/del esfuerzo de torsión [34][35].

Hoy en día, las capacidades de visión a menudo se integran en una celda de trabajo robótica. Se pueden montar una o más cámaras en el robot (*configuración ojo a mano*) o en la celda de trabajo (*configuración ojo a mano*) para realizar mediciones y supervisar la escena (Figure 9.4-7). Por lo tanto, pueden proporcionar información sobre los objetos que se van a procesar y los dispositivos y el entorno circundantes.





**Figura 9.4-6** A Compensador RCC por ATI Industrial Automation: a) Esquema conceptual\*; b) Imagen del dispositivo†. Derechos de imagen: ATI Industrial Automation.



**Figura 9.4-7** Configuraciones del sistema de visión (caso simple de una sola cámara): a) ojo a mano; b) ojo a mano. Derechos de imagen: Autores.

En el enfoque más simple, el robot recibe el mando de control de movimiento, pero explotando la información proporcionada por la cámara, implementando la llamada *estrategia de mirar y mover*. Como ejemplo, considere un robot que tiene que elegir un objeto en el área de trabajo cuya vista es proporcionada por una cámara. La cámara identifica el objeto y detecta su pose que, mediante una calibración adecuada, se puede expresar en el sistema de coordenadas del robot. Esta información se envía al controlador del robot que ordena al robot que mueva y recoja el objeto.

Cuando la información visual se utiliza para cerrar el bucle de control, dicho control basado en la retroalimentación visual se refiere a *servo control visual* [36]. En otras palabras, "*servo control visual se refiere al uso de datos de visión por computadora para controlar el movimiento de un robot*" [37].

\* Fuente: [https://www.ati-ia.com/Products/Compliance/Compensator\\_product\\_desc.aspx](https://www.ati-ia.com/Products/Compliance/Compensator_product_desc.aspx)

† Fuente: [https://www.ati-ia.com/products/compliance/Compensator\\_ModelDetails.aspx?id=9116-001-A](https://www.ati-ia.com/products/compliance/Compensator_ModelDetails.aspx?id=9116-001-A)

Los dos enfoques principales para el servo visual son el *servo visual basado en la posición* y el *servo visual basado en la imagen* (o enfoques híbridos). En ambos casos, el desafío es mover la herramienta del robot a una pose específica en relación con el objeto objetivo. Con el servo control visual basado en la posición, las mediciones visuales se utilizan para reconstruir la pose relativa del objeto con respecto al robot, mientras que el *servoing* visual basado en la imagen se basa en la comparación de los parámetros de características de la imagen del objeto entre la pose actual y la deseada.

Para comprender mejor los dos enfoques, considere el caso de la configuración ojo a mano, donde la herramienta robot está en una postura relativa fija a la cámara y el problema se refiere a mover la cámara.

En el servo control visual basado en la posición, se utiliza un modelo geométrico del objeto y los parámetros de la cámara para estimar la pose del objeto en relación con la cámara. Entonces es posible calcular el error de pose y ordenar al robot que se mueva para corregir la pose.

En el servo control visual basado en imágenes, no es necesario un modelo del objeto, pero es necesario detectar características específicas (por ejemplo, coordenadas de puntos) en la imagen del objeto adquirido por la cámara. La posición de los puntos en la imagen depende de la posición de la cámara en relación con el objeto, de modo que si se mueve la cámara, los puntos se mueven en la imagen. Cuando la cámara está en la postura deseada, los puntos aparecen en posiciones conocidas en la imagen, de lo contrario, aparecen en diferentes posiciones. El *servoing* visual basado en imágenes compara las características de puntos deseadas y actuales y calcula la velocidad deseada de la cámara para mover los puntos donde sea necesario en la imagen, por lo tanto, la cámara a la pose deseada.

Los algoritmos de procesamiento de imágenes son esenciales para extraer las características de imagen necesarias para controlar el robot.

## 9.5 Inteligencia Artificial

La robótica es uno de los mercados de alta tecnología de más rápido crecimiento, debido principalmente a la introducción generalizada de robots en la producción, y al creciente sector de servicios y cuidados domiciliarios. Los dispositivos robóticos reales han alcanzado un alto nivel de madurez y fiabilidad en términos de estructuras mecánicas, sensores y actuadores, y técnicas de control para resolver tareas básicas de percepción, navegación y manipulación cuando se trabaja en entornos estructurados y bien definidos.

Sin embargo, para cumplir con los paradigmas I4.0, los robots deberán ser cada vez más versátiles, abordando una serie de tareas en entornos no estructurados y abiertos. Por lo tanto, deberán estar equipados con mayores habilidades cognitivas, como la representación del conocimiento, la planificación, el aprendizaje, la adaptación y la interacción humano-robot natural.

La inteligencia artificial es una rama de la informática relacionada con la programación y el diseño de sistemas de hardware y software, permitiendo que las máquinas tengan habilidades típicas de los seres humanos y animales, como las percepciones visuales, espacio-tiempo y decisión.

Por lo tanto, el concepto de inteligencia se define no solo como la capacidad de calcular datos abstractos, sino que también incluye todas las formas diferentes reconocidas por la teoría de las inteligencias múltiples de Gardner [37] que van desde la inteligencia espacial hasta la social, desde la cinestésica hasta la introspectiva.

Un sistema inteligente se crea tratando de recrear una o más de estas diferentes formas de inteligencia. Aunque a menudo se definen como exclusivamente humanos o naturales, pueden simplificarse a comportamientos particulares reproducibles por algunas máquinas.

La inteligencia artificial ya se explota en gran medida en la vida cotidiana. Por ejemplo, las diversas herramientas de reconocimiento de voz que se utilizan regularmente en muchas aplicaciones, desde teléfonos inteligentes hasta sistemas de seguridad, se basan en algoritmos de aprendizaje automático, que son una forma de IA.

Otra aplicación popular del aprendizaje automático y el paradigma de inteligencia artificial se encuentra en el sector automotriz, donde los sistemas avanzados de cambio de velocidad en automóviles de conducción semiautónomos hacen uso de algoritmos de IA basados en la lógica difusa [39]. Además, la IA está desempeñando un papel clave en el desarrollo de automóviles de conducción autónoma, aunque aún deben abordarse muchos desafíos técnicos y éticos para tener un vehículo autónomo confiable y seguro.

En el campo médico, la inteligencia artificial utiliza principalmente redes neuronales, con aplicaciones en el análisis de los latidos del corazón, en el diagnóstico de algunas formas de cáncer y en la creación de nuevos fármacos terapéuticos. Recientemente, se han utilizado técnicas de IA para desarrollar herramientas para predecir la pandemia de coronavirus y la detección y diagnóstico tempranos de infecciones [40].

Otros sectores en los que la inteligencia artificial se usa regularmente son el mercado de valores, el reconocimiento de imágenes (incluido el reconocimiento facial), el marketing en redes sociales y la robótica. Además, los sistemas inteligentes también se utilizan para mejorar aún más muchos sectores de la tecnología de la información en sí, incluida la demostración de la corrección de los algoritmos y la creación de nuevos métodos de aprendizaje.

### 9.5.1 Resumen histórico

En 1954, George Devol construyó el primer robot programable, llamado Unimate. Era un brazo manipulador hidráulico que podía realizar tareas repetitivas. Fue comprado por General Motors en 1961 para su uso en líneas de ensamblaje de automóviles para automatizar los procesos de metalurgia y soldadura. Aunque a la luz del rendimiento actual de la IA, su grado de inteligencia es muy rudimentario,

en el momento de su invención fue considerado por muchos como uno de los pocos productos de IA con valor comercial.

Desde estos primeros desarrollos, se han realizado trabajos innovadores, mejorando específicamente las capacidades robóticas para la detección y la percepción (principalmente a través de la visión artificial, pero también agregando retroalimentación de fuerza, habilidades táctiles, sensores de proximidad, etc.) aumentando así las capacidades de las máquinas para interactuar con el entorno circundante. Ya a finales de la década de 1970 se desarrollaron robots humanoides que podían apilar bloques. Las nuevas mejoras en el hardware de las TIC y las tecnologías de IA impulsaron el desarrollo de sistemas robóticos cada vez más avanzados.

Desde la acuñación del término AI en 1956 hasta principios de la década de 1970, el entusiasmo por la IA dio a luz a varios subcampos de investigación, que constituyen la base de la teoría moderna de la IA: sistemas basados en reglas, aprendizaje automático, procesamiento del lenguaje natural (NLP), redes neuronales superficiales y profundas, procesamiento de imágenes y visión por computadora, reconocimiento de texto hablado. Sin embargo, las aplicaciones prácticas de los programas basados en IA todavía eran muy poco comunes y se limitaban en gran medida a resolver problemas rudimentarios, principalmente debido a las limitaciones en la potencia de cálculo disponible.

Además, la mayoría de las primeras predicciones sobre el futuro de las máquinas fuertes artificialmente inteligentes aún no se han realizado, lo que hace que los inversores sean escépticos y reacios a invertir más fondos. Esto dio lugar a una fase de busto de AI. La mayor parte del número limitado de invenciones de este período, como la propagación posterior y las redes neuronales recurrentes, se pasó por alto en gran medida y se empleó un esfuerzo sustancial para redescubrirlas en las décadas siguientes.

Los recientes avances en las tecnologías de la información y la comunicación dieron nueva vida a la investigación en IA, ya que el hardware y la conectividad de red se volvieron más baratos y rápidos; los sistemas paralelos y distribuidos se volvieron prácticos, y enormes cantidades de datos (Big Data) se almacenaron fácilmente y luego se disponían para entrenar sistemas de IA.

### **9.5.2 Técnica**

El uso de algoritmos capaces de reproducir el razonamiento de seres humanos o animales en diferentes situaciones permitió a los sistemas inteligentes mejorar aún más su conjunto de habilidades de comportamiento. Los esfuerzos de investigación se centraron en el desarrollo de algoritmos que podrían imitar diferentes comportamientos según los estímulos ambientales (la habilidad a menudo llamada conciencia del contexto).

Por lo tanto, estos complejos algoritmos implementados dentro de sistemas inteligentes son capaces de tomar decisiones o elecciones de acuerdo con el contexto en el que se aplican. En el caso de los sistemas de vehículos inteligentes,

por ejemplo, un automóvil sin conductor puede decidir en caso de peligro si dirigir o frenar de acuerdo con la situación. Por ejemplo, dependiendo de si la información enviada por los diversos sensores, como la velocidad y la dirección del automóvil, el tamaño y la posición de otros objetos, y las condiciones del suelo y el clima, calcula si el frenado o la dirección tienen un mayor porcentaje de seguridad para el conductor, los pasajeros y los peatones. El conocimiento básico originalmente incrustado en el sistema es que expandido, creado a través de la experiencia.

Ha crecido un sector específico, centrado en la representación del conocimiento, que estudia todos los mecanismos del razonamiento humano y define métodos para hacer que este conocimiento sea comprensible para las máquinas, a través del lenguaje y controles cada vez más precisos y detallados. La transferencia de conocimientos humanos existentes a máquinas requiere transferir no solo conocimientos de dominio, sino también experiencias. Esto agrega la posibilidad de comprender nueva información explotando los conocimientos ya presentes en el sistema de inicio.

Esta información se proporciona a la máquina de varias maneras, las más importantes de las cuales son las basadas en la *Teoría de los Lenguajes Formales* (que estudia los aspectos sintácticos de los lenguajes estructurados) y la *Teoría de la Decisión*, para identificar decisiones óptimas bajo circunstancias específicas.

Uno de los principales pasos adelante en el desarrollo de la IA se dio cuando se crearon algoritmos que eran capaces de aprender a través de la experiencia al igual que los humanos. El aprendizaje hace que el sistema sea capaz de mejorar la capacidad de la máquina para actuar y tomar decisiones.

Desarrollar algoritmos que puedan aprender de sus experiencias y errores es esencial para crear sistemas inteligentes que operan en situaciones para las que los programadores no pueden prever todas las posibilidades futuras.

A través del *aprendizaje automático*, también discutido en el Capítulo 5, una máquina puede aprender a realizar una determinada acción, incluso si esta acción nunca se ha programado entre las posibles acciones. Por ejemplo, un robot puede realizar una clavija en la operación de ensamblaje de orificios incluso si la forma y la posición del orificio difieren de las que fue entrenado con, o puede reconocer un objeto a recoger incluso si esa imagen exacta no se almacenó previamente en su memoria.

El aprendizaje automático es probablemente la rama más común de la inteligencia artificial considerada en la cultura dominante, y constituyó la base de varias obras literarias exitosas. Sin embargo, la investigación aún está lejos de los escenarios representados en las películas y la literatura de ciencia ficción, y se desarrolla a lo largo de caminos teóricos y prácticos en la teoría del aprendizaje computacional y el reconocimiento de patrones.

El aprendizaje automático fue permitido por el desarrollo de *redes neuronales artificiales*, es decir, un modelo matemático particular inspirado en cerebros y neuronas. El nombre de *red neuronal* deriva del hecho de que este modelo matemático se caracteriza por una serie de interconexiones entre neuronas artificiales, y estas conexiones son necesarias para los diferentes cálculos. Al igual

que las redes neuronales biológicas, una red neuronal artificial también tiene la característica de ser adaptable a las diferentes necesidades derivadas de la diferente información obtenida en las diferentes fases de aprendizaje.

## 9.6 Cuestiones éticas

A medida que aumenta el grado de autonomía y capacidad de toma de decisiones de los sistemas automáticos, es esencial definir un código de conducta, una lista de normas que deben respetarse para evitar comportamientos imprevistos. Un caso típico es el de los vehículos guiados autónomos, como AGV industriales utilizados para aplicaciones logísticas, drones y automóviles autónomos. Es muy probable que, en un futuro próximo, estos conceptos se extiendan también a los robots que actúan en el entorno industrial.

La *ética del robot* se define como el conjunto de reglas y el código de conducta que los diseñadores integran en el sistema de control del robot.

El primer conjunto de reglas (las conocidas *tres leyes de la robótica*) fue definido por el escritor de ciencia ficción Isaac Asimov [40], y la comunidad científica considera que todavía son válidas hoy en día:

1. Un robot no puede dañar a un ser humano o, a través de la inacción, permitir que un ser humano sufra daños.
2. Un robot debe obedecer las órdenes dadas por los seres humanos, excepto cuando tales órdenes entren en conflicto con la Primera Ley.
3. Un robot debe proteger su propia existencia siempre que dicha protección no entre en conflicto con la Primera o Segunda Leyes.

Más tarde, añadió también una *Ley 0*: Ningún robot puede dañar a la humanidad o, a través de la inacción, permitir que la humanidad haga daño.

Sin embargo, estas leyes no son suficientes y puede haber escenarios en los que no puedan obedecerse plenamente.



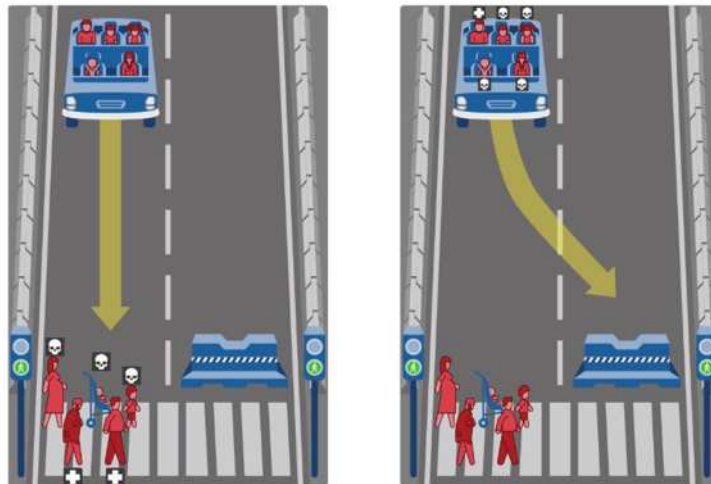
**Figura 9.6-1** Configuración autónoma del coche<sup>‡</sup>.

<sup>‡</sup> Image design by Freepik.

Las tres leyes implican que un robot tiene suficiente inteligencia (en términos de percepción y capacidades cognitivas) para tomar también decisiones morales en situaciones complejas, lo que todavía no es el caso. Por lo tanto, no pueden proporcionar una base práctica para escribir un código de conducta de robot. Un ejemplo de un conjunto de reglas fácilmente incrustables en un sistema robótico que fue propuesto por científicos robo-éticos es:

1. No matar
2. No causar dolor
3. No desactivar
4. No prive de la libertad
5. No engañar
6. Mantenga su promesa
7. No hagas trampa
8. Obedecer la ley
9. Haga su deber

En cuanto a la ética, es necesario abordar varios dilemas morales.



**Figura 9.6-2** dilema moral del coche sin conductor **Error! Reference source not found..**

Figure 9.6-2 muestra un dilema moral típico, que apenas podría estar incrustado en el código de software de la máquina: sabiendo que solo hay dos eventos posibles: chocar contra las personas que cruzan la calle y matar a tres de ellos o chocar el automóvil contra la barrera y matar a cuatro pasajeros, ¿qué alternativa debería elegirse? Obviamente, esta sería una decisión difícil también para un conductor humano, con muchos factores diferentes a considerar.

Varios proyectos de investigación están en curso en el campo y muchos otros ejemplos de dilemas morales se pueden encontrar en <http://moralmachine.mit.edu/>, una plataforma experimental en línea diseñada para explorar los dilemas morales que enfrentan los vehículos autónomos y para recopilar opiniones humanas sobre cómo las máquinas deben tomar decisiones cuando se enfrentan a estos dilemas morales.

Estos temas también se han utilizado como un principio de referencia en el contexto del estudio encargado por la Comisión de Asuntos Jurídicos del Parlamento Europeo "para evaluar y analizar, desde una perspectiva jurídica y ética, una serie de futuras normas europeas de Derecho civil en robótica". En febrero de 2017 se adoptó la subsiguiente Resolución del Parlamento Europeo «Leyes civiles en robótica», que abarca una amplia gama de ámbitos diferentes, incluidas las normas de responsabilidad, cuestiones éticas, normalización, seguridad, protección de datos y derechos de propiedad intelectual, vehículos autónomos, drones, robots de cuidado, robots médicos, mejora humana y educación y empleo y un código de conducta ética para ingenieros. Además, propone una Agencia de Robótica e Inteligencia Artificial de la UE.

La reacción de las empresas de fabricación de robots fue bastante rápida, mostrando ansiedad de que esta regulación podría obstaculizar el progreso y amenazar la competitividad de toda la industria manufacturera. También se han llevado a cabo varios estudios sociales para evaluar el impacto de estas tecnologías en rápida evolución en el mercado de trabajo, y todavía hay un fuerte debate en los periódicos, las redes sociales y las organizaciones internacionales.

De hecho, el 22 de mayo de 2019, la Organización para la Cooperación y el Desarrollo Económicos adoptó la Recomendación del Consejo de la OCDE sobre Inteligencia Artificial **Error! Reference source not found.**, que establece los principios para la administración responsable de sistemas de Inteligencia Artificial confiables y recomendaciones a los gobiernos nacionales para las políticas nacionales y la cooperación internacional. Estos principios se basan en: crecimiento inclusivo, desarrollo sostenible y bienestar, valores centrados en el ser humano y equidad, transparencia y explicabilidad, solidez, seguridad y seguridad; rendición de cuentas.

## 9.7 Conclusiones

Las innovaciones tecnológicas de la Industria 4.0 permiten una nueva generación de fábricas donde los robots industriales realizan de forma autónoma todas las tareas, mientras que los trabajadores tienen principalmente un papel de supervisión. Gracias a las capacidades de sensorización mejoradas y al control en tiempo real, la cooperación entre humanos y robots se vuelve más fácil y segura, mejorando la productividad, la calidad de producción y la flexibilidad. Los sistemas ciber-físicos, provistos de inteligencia artificial, no sólo monitorean todo el proceso de



producción, sino que realizan predicciones y toman decisiones con el fin de mejorar la eficiencia de la producción.

En las últimas décadas se han logrado muchos progresos, pero todavía hay que afrontar varios desafíos, tanto en el mundo físico (hardware) como en el cibernético (software), mejorando sensores, actuadores, controladores, sistemas de almacenamiento y análisis de datos, y algoritmos de inteligencia artificial, con el fin de lograr la nueva generación de fábricas concebidas por el paradigma de la Industria 4.0. La interconexión requiere un *sistema de comunicación estándar*, por lo que se proponen que los lenguajes propietarios sean sustituidos por soluciones de código abierto; la colaboración humano-robot necesita de *habilidades avanzadas de percepción* para trabajar de forma segura en espacios no estructurados y los *aspectos psicosociales* de esta colaboración, desde el punto de vista de los trabajadores, deben ser considerados sabiamente; las máquinas inteligentes capaces de tomar decisiones introducen serios *problemas éticos*, por lo que leyes convenientes y apropiadas deben ser definidas e implementadas en los robots; los CPS están expuestos a amenazas a la seguridad y la transmisión y el intercambio de datos requiere nuevos niveles de *ciberseguridad y privacidad*; la modularidad y configurabilidad necesarios para los productos totalmente personalizados requiere nuevos *modelos de negocio*; todas las transformaciones necesarias requieren los correspondientes cambios en las habilidades de la carga de trabajo y las estructuras organizativas, que se pueden lograr con una transformación en el *sistema educativo*, ofreciendo no sólo nuevos contenidos sino también nuevas metodologías para la nueva generación de empleados. La industria 4.0 es una revolución industrial que evoluciona a un ritmo exponencial con un enorme impacto en la calidad de vida de las poblaciones de todo el mundo. Un conocimiento responsable de tal impacto es esencial para sacar el máximo provecho de este avance, ya que también podría dar lugar a una mayor desigualdad. Los beneficios inmediatos serán principalmente para los innovadores e inversores, el mercado laboral se verá fuertemente afectado, aumentando la demanda de trabajadores altamente calificados y proporcionando pocas ofertas a los trabajadores con menor educación y habilidades. Las interconexiones digitales también acortarán las distancias entre las personas, proporcionando nuevas oportunidades para los intercambios interculturales, mejorando la comprensión y la cohesión. Sin embargo, también las expectativas poco realistas y las ideologías extremas se extenderán más rápido y sin control, accediendo a personas más vulnerables que también podrían ser víctimas de violaciones de privacidad y del robo de datos. Por lo tanto, la oportunidad que ofrece la cuarta revolución industrial tiene que ser globalmente compartida y conformada hacia un futuro con objetivos y valores comunes, para beneficiar a las personas indefensas en lugar de aumentar la brecha entre las clases sociales.

**Términos de acceso (Open Access)** Este capítulo se distribuye bajo los términos de la Licencia Internacional Creative Commons Attribution 4.0 (<http://creativecommons.org/licenses/by/4.0/>), que permite el uso, duplicación, adaptación, distribución y reproducción en cualquier medio o formato, siempre que dé el crédito apropiado al autor o autores originales y la fuente, se menciona y proporciona un enlace a la licencia Creative Commons y se indica cualquier cambio realizado.

Las imágenes u otro material de terceros en este capítulo están incluidos en la licencia Creative Commons del trabajo, a menos que se indique lo contrario en la línea de crédito de cada material; Si dicho material no está incluido en la licencia Creative Commons de la obra y la acción respectiva no está permitida por la normativa legal, los usuarios deberán obtener permiso del titular de la licencia para duplicar, adaptar o reproducir dicho material

## 9.8 Referencias

- [1] N. Jazdi, “Cyber physical systems in the context of Industry 4.0”, in Proceedings of the IEEE International Conference on Automation, Quality and Testing, Robotics, Cluj-Napoca, Romania, 22–24 May, pp. 1- 4, 2014.
- [2] E. Geisberger, & M. Broy, (Eds.). agendaCPS: Integrierte Forschungsagenda Cyber-Physical Systems Vol. 1. Springer-Verlag.; ISBN 978-3-642-29098-5, 2012
- [3] D. Zuehlke, “SmartFactory Towards a factory-of-things”, *Annual Reviews in Control*, vol.34, pp.129-138, 2010.
- [4] A. Khalid et al. “A methodology to develop collaborative robotic cyber-physical systems for production environments”, *Logistic Research* 9:23, 2016.
- [5] I. Navarro and F. Matía, “An introduction to swarm robotics”. *International Scholarly Research Notices*, 2013.
- [6] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klapotcz, S. Magnenat, J.C. Zufferey, D. Floreano, and A. Martinoli, “The e-puck, a robot designed for education in engineering”, in Proceedings of the 9th conference on autonomous robot systems and competitions Vol. 1, No. CONF, pp. 59-65, IPCB: Instituto Politécnico de Castelo Branco, 2009.
- [7] M.H.M. Alkilabi, A. Narayan, C. Lu, and E. Tuci, Evolving group transport strategies for e-puck robots: moving objects towards a target area. In Distributed Autonomous Robotic Systems, pp. 503-516 Springer, Cham. 2018.
- [8] Alkilabi, M.H.M., Narayan, A. y Tuci, E., 2017. Cooperative object transport with a swarm of e-puck robots: robustness and scalability of evolved collective strategies”, *Swarm intelligence*, 11 (3-4), pp.185-209.
- [9] F. Perez-Diaz, R. Zillmer, and R. Groß, “Control of synchronization regimes in networks of mobile interacting agents”,. *Physical Review Applied*, 7(5), p.054002, 2017.

- [10] <http://www.e-puck.org/>
- [11] <https://www.youtube.com/watch?v=t8wbicHdRFk>
- [12] <https://www.youtube.com/watch?v=E7jRYvEFytE>
- [13] [https://www.youtube.com/watch?v=4DKrcpa8Z\\_E](https://www.youtube.com/watch?v=4DKrcpa8Z_E)
- [14] Siciliano, B., Sciavicco, L., Villani, L., Oriolo, G. *Robotics - Modelling, Planning and Control*, Capítulo 4, Springer-Verlag London Limited 2010. ¿Qué puedes encontrar en Neodigit
- [15] Legnani, G., Fassi, I. *Robotica Industriale*, Capítulo 9, CittàStudi, 2019.
- [16] Hägele, M., Nilsson, K., Pires, J. N., Bishoff, R. *Industrial Robotics*, in *Springer Handbook of Robotics*, Capítulos 54, Siciliano, B., Khatib, O. (Eds.), 2ª edición, 2016. Sistema abierto.
- [17] Siciliano, B., Sciavicco, L., Villani, L., Oriolo, G. *Robotics - Modelling, Planning and Control* Capítulo 6, Springer-Verlag London Limited 2010. ¿Qué puedes encontrar en Neodigit
- [18] Legnani, G., Fassi, I. *Robotica Industriale*, Capítulo 11, CittàStudi, 2019.
- [19] <https://new.abb.com/products/robotics/robotstudio>
- [20] <https://es.mitsubishielectric.com/fa/es/products/industrial-robots-melfa/engineering-software/rt-toolbox3-pro/rt-toolbox3#>
- [21] <https://robodk.com/>
- [22] <https://www.robotmaster.com/en/>
- [23] [https://www.delfoi.com/web/products/delfoirobotics/es\\_ES/off-line\\_ENG/](https://www.delfoi.com/web/products/delfoirobotics/es_ES/off-line_ENG/)
- [24] [http://www.fastsuite.com/en\\_EN/fastsuite/why-fastsuite.html](http://www.fastsuite.com/en_EN/fastsuite/why-fastsuite.html)
- [25] <http://octopuz.com/>
- [26] <http://wiki.ros.org/ROS/Introduction>
- [27] <https://rosindustrial.org/about/description/>
- [28] Siciliano, B., et al., *Robotics - Modelling, Planning and Control*, Capítulo 8, Springer-Verlag London Limited 2010. ¿Qué puedes encontrar en Neodigit
- [29] G. Legnani, I. Fassi, *Robotica Industriale*, Chapter 10, CittàStudi, 2019.
- [30] Villani, Luigi et al., *Force Control*, Chapter 9, in *Springer Handbook of Robotics*, Siciliano, B., Khatib, O. (Eds.), 2nd edition, 2016. Sistema abierto.
- [31] Siciliano, Bruno, et al., *Robotics - Modelling, Planning and Control I*, Capítulo 9, Springer-Verlag London Limited 2010. ¿Qué puedes encontrar en Neodigit
- [32] Kevin M. Lynch and Frank C. Park, *Modern Robotics: Mechanics, Planning, and Control*, capítulo 11, Cambridge University Press, 2017, ISBN 9781107156302.
- [33] Hannaford, Blake, et al., *Haptics*, Chapter 42, in *Springer Handbook of Robotics*, Siciliano, B., Khatib, O. (Eds.), 2nd edition, 2016. Sistema abierto.
- [34] Niemeyer, Günte, et al., *Telerobotics*, Chapter 43, in *Springer Handbook of Robotics*, Siciliano, B., Khatib, O. (Eds.), 2nd edition, 2016. Sistema abierto.
- [35] Chaumette, François, et al., *Visual Servoing*, Chapter 34, in *Springer Handbook of Robotics*, Siciliano, B., Khatib, O. (Eds.), 2nd edition, 2016. Sistema abierto.

- [36] Siciliano, Bruno, et al., *Robotics - Modelling, Planning and Control*, Chapter 10, Springer-Verlag London Limited 2010. ¿Qué puedes encontrar en Neodigit
- [37] H. Gardner, *Frames of mind: The Theory of Multiple Intelligences*, 1983
- [38] Fuzzy Logic (FL) es un método de razonamiento que se asemeja al proceso de toma de decisiones humanas, trabajando en los niveles de posibilidades de entrada para lograr la salida definida, ayudando a lidiar con las incertidumbres en la ingeniería. Se puede implementar tanto a nivel de hardware como de software.
- [39] Menni, C., *et al.* (2020) Real-time tracking of self-reported symptoms to predict potential COVID-19. *Nature Medicine*. [doi.org/10.1038/s41591-020-0916-2](https://doi.org/10.1038/s41591-020-0916-2).
- [40] I. Asimov, *Runaround*, 1942
- [41] E. Awad, S. Dsouza, R. Kim, J. Schulz, J. Henrich, A. Shariff, J.-F. Bonnefon, I. Rahwan, “The Moral Machine experiment”, *Nature*, 563, pp.59–64, 2018.
- [42] OECD, Recommendation of the Council on Artificial Intelligence, OECD/LEGAL/0449



Co-funded by the  
Erasmus+ Programme  
of the European Union



"El apoyo de la Comisión Europea para la producción de esta publicación no constituye una aprobación del contenido que refleja únicamente las opiniones de los autores, y la Comisión no se hace responsable del uso que pueda hacerse de la información contenida en el mismo"

Sistema de formación en fabricación Digital para Pymes (Digit-T)  
Referencia del proyecto: 2017-1-UK01-KA202-036807