

Facultad de Informática y Matemática

Oscar Lucero Moya

Holguín

Estudio sobre los lenguajes de programación para la robótica

2001

Índice

Índice

Introducción

Origen de la palabra robot y su significado

A través de la historia el hombre a soñado con seres capaces de realizar movimientos y hasta comportarse parecido a animales y hasta seres humanos; en el fondo el ser humano lo que piensa es librarse de tareas indeseables, peligrosas o demasiados tediosas. La primera vez que se hablo de estos seres utilizando el término por el que es hoy mundialmente conocido, robot, fue en 1923 por el escritor Karel Capek en su comedia R.U.R..(Rossum`s Universal Robots), palabra que proviene del término checo *robotnik* que significa siervo.

Hoy la palabra robot tiene diferentes significados:

- La del Robot Institute of América: un manipulador multifuncional y reprogramable, diseñado para mover materiales piezas, herramientas o dispositivos especiales, mediante movimientos programables y variables que permitan llevar a cabo diversas tareas.
- La del Oxford English dictionary: un aparato mecánico que se parece y hace el trabajo de un ser humano.

cronograma sobre la historia de la robótica

FECHA	DESARROLLO
SigloXVIII.	A mediados del J. de Vaucanson construyó varias muñecas mecánicas de tamaño humano que ejecutaban piezas de música
1801	J. Jaquard invento su telar, que era una máquina programable para la urdimbre
1805	H. Maillardet construyó una muñeca mecánica capaz de hacer dibujos.
1946	El inventor americano G.C Devol desarrolló un dispositivo controlador que podía registrar señales eléctricas por medio magnéticos y reproducirlas para accionar un máquina mecánica. La patente estadounidense se emitió en 1952.
1951	Trabajo de desarrollo con teleoperadores (manipuladores de control remoto) para manejar materiales radiactivos. Patente de Estados Unidos emitidas para Goertz (1954) y Bergsland (1958).
1952	Una máquina prototipo de control numérico fue objetivo de demostración en el Instituto Tecnológico de Massachusetts después de varios años de desarrollo. Un lenguaje de programación de piezas denominado APT (Automatically Programmed Tooling) se

	desarrolló posteriormente y se publicó en 1961.
1954	El inventor británico C. W. Kenward solicitó su patente para diseño de robot. Patente británica emitida en 1957.
1954	G.C. Devol desarrolla diseños para Transferencia de artículos programada. Patente emitida en Estados Unidos para el diseño en 1961.
1959	Se introdujo el primer robot comercial por Planet Corporation. estaba controlado por interruptores de fin de carrera.
1960	Se introdujo el primer robot `Unimate", basada en la transferencia de articulaciones programada de Devol. Utilizan los principios de control numérico para el control de manipulador y era un robot de transmisión hidráulica.
1961	Un robot Unimate se instaló en la Ford Motors Company para atender una máquina de fundición de troquel.
1966	Trallfa, una firma noruega, construyó e instaló un robot de pintura por pulverización.
1968	Un robot móvil llamado `Shakey" se desarrollo en SRI (standford Research Institute), estaba provisto de una diversidad de sensores así como una cámara de visión y sensores táctiles y podía desplazarse por el suelo.
1971	El `Standford Arm", un pequeño brazo de robot de accionamiento eléctrico, se desarrolló en la Standford University.
1973	Se desarrolló en SRI el primer lenguaje de programación de robots del tipo de computadora para la investigación con la denominación WAVE. Fue seguido por el lenguaje AL en 1974. Los dos lenguajes se desarrollaron posteriormente en el lenguaje VAL comercial para Unimation por Víctor Scheinman y Bruce Simano.
1974	ASEA introdujo el robot Irb6 de accionamiento completamente eléctrico.
1974	Kawasaki, bajo licencia de Unimation, instaló un robot para soldadura por arco para estructuras de motocicletas.
1974	Cincinnati Milacron introdujo el robot T3 con control por computadora.
1975	El robot `Sigma" de Olivetti se utilizó en operaciones de montaje, una de las primitivas aplicaciones de la robótica al montaje.
1976	Un dispositivo de Remopte Center Compliance (RCC) para la inserción de piezas en la línea de montaje se desarrolló en los laboratorios Charles Stark Draper Labs en estados Unidos.
1978	El robot T3 de Cincinnati Milacron se adaptó y programó para realizar operaciones de taladro y circulación de materiales en componentes de aviones, bajo el patrocinio de Air Force ICAM (Integrated Computer- Aided Manufacturing).
1978	Se introdujo el robot PUMA (Programmable Universal Machine for Assambly) para tareas de montaje por Unimation, basándose en diseños obtenidos en un estudio de la General Motors.
1979	Desarrollo del robot tipo SCARA (Selective Compliance Arm for Robotic Assambly) en la Universidad de Yamanashi en Japón para montaje. Varios robots SCARA comerciales se introdujeron hacia 1981.
1980	Un sistema robótico de captación de recipientes fue objeto de demostración en la Universidad de Rhode Island. Con el empleo de visión de máquina el sistema era capaz de captar piezas en orientaciones aleatorias y posiciones fuera de un recipiente.
1981	Se desarrolló en la Universidad de Carnegie- Mellon un robot de impulsión directa. Utilizaba motores eléctricos situados en las articulaciones del manipula dor sin las transmisiones mecánicas habituales empleadas en la mayoría de los robots.
1982	IBM introdujo el robot RS-1 para montaje, basado en varios años de desarrollo interno. Se trata de un robot de estructura de caja que utiliza un brazo constituido por tres

	dispositivos de deslizamiento ortogonales. El lenguaje del robot AML, desarrollado por IBM, se introdujo también para programar el robot SR-1.
1983	Informe emitido por la investigación en Westinghouse Corp. bajo el patrocinio de National Science Foundation sobre un sistema de montaje programable adaptable (APAS), un proyecto piloto para una línea de montaje automatizada flexible con el empleo de robots.
1984	Robots 8. La operación típica de estos sistemas permitía que se desarrollaran programas de robots utilizando gráficos interactivos en una computadora personal y luego se cargaban en el robot.

PROPIEDADES características de los robots

- Versatilidad:

Potencialidad estructural de ejecutar tareas diversas y/o ejecutar una misma tarea de forma diversa. Esto impone al robot una estructura mecánica de geometría variable.

- Autoadaptabilidad al entorno:

Significa que un robot debe, por sí solo, alcanzar su objetivo(ejecutar su tarea) a pesar de las perturbaciones imprevistas del entorno a lo largo de la ejecución de su tarea. Esto supone que el robot sea consciente de su entorno y que por lo tanto posea sentidos artificiales.

el robot y su funcionamiento

Un robot operacional puede estar constituido por cuatro entidades unidas entre sí(ver anexo no.4 Descripción de un robot en funcionamiento):

- Sistema mecánico articulado dotado de sus motores(eléctricos, hidráulicos o neumáticos) que arrastran a las articulaciones del robot mediante las transmisiones(cables, cintas, correas con muescas). Para conocer en todo instante la posición de las articulaciones se recurre a los captadores(codificadores ópticos) que se denominan propioceptivos. Estos dan el valor a las articulaciones, que no es más que la configuración o el estado del robot.
- El entorno es el universo en que está sumergida la primera entidad. Si los robots están sobre un puesto fijo se reduce al espacio alcanzable por el robot. En él el robot puede encontrar obstáculos que ha de evitar y objetos de interés, o sea los objetos con los que tiene que actuar. Por todo esto existe interacción entre la parte física y el entorno. Mediante los captadores exteroceptivos(cámaras, detectores de fuerzas, detectores de proximidad, captadores táctiles) se toman informaciones sobre el entorno.
- Las tareas a realizar es el trabajo que se desea que haga el robot. La descripción de estas tareas se hace mediante lenguajes que pueden ser a través de los gestos, en el que se le enseña al robot lo que se debe hacer; orales, se le habla; por escrito en el que se le escriben las instrucciones en un lenguaje compatible con el robot.
- El cerebro del robot es el órgano de tratamiento de la información. Este puede ser desde un autómata programable para los menos avanzados hasta un miniordenador numérico o microprocesador para los más avanzados. El cerebro, es el que tiene el papel principal, contiene en sus memorias:
 - Un modelo del robot físico: las señales de excitación de los accionadores y los desplazamientos que son consecuencia de ellas.
 - Un modelo del entorno: descripción de lo que se encuentra en el espacio que puede alcanzar.
 - Programas: permite comprender las tareas que se le pide que realice. Algoritmos de control.

clasificación de los robots

CRITERIO	CLASIFICACIÓN
Geometría	Se basa en la forma del área de trabajo producida por el brazo del robot: rectangular, cilíndrica o esférica.
Configuraciones	La configuración polar utiliza coordenadas polares para especificar cualquier posición en términos de una rotación sobre su base, un ángulo de elevación y una extensión lineal del brazo.
	La configuración cilíndrica sustituye un movimiento lineal por uno rotacional sobre su base, con los que se obtiene un medio de trabajo en forma de cilindro.
	La configuración de coordenadas cartesianas posee tres movimientos lineales, y su nombre proviene de las coordenadas cartesianas, las cuales son más adecuadas para describir la posición y movimiento del brazo. Los robots cartesianos a veces reciben el nombre de XYZ, donde las letras representan a los tres ejes del movimiento.
Grados de libertad	Consiste en contar el número de grados de libertad que tengan. Se considera un grado de libertad cada eje a lo largo del cual se puede mover el brazo de un robot.
Técnica de control	Ensamblaje
	No ensamblaje: soldar, pintar, revestir, manejo de materiales y carga y descarga de maquinaria.
	Lazo cerrado: se monitorea continuamente la posición del brazo del robot mediante un sensor de posición, y se modifica la energía que se manda al actuador de tal forma que el movimiento del brazo se obedece al camino deseado, tanto en dirección como en velocidad. Éste control se puede usar cuando la tarea que se ha de llevar a cabo está dirigida mediante un camino definido por la misma pieza, tal como sería soldar, revestir y ensamblar.
	En un sistema de lazo abierto, el controlador no conoce la posición de la herramienta mientras el brazo se mueve de un punto a otro. Éste tipo de control es muy usado cuando el movimiento que debe seguir el brazo se encuentra determinado previamente, al ser grabado con anterioridad y reproducido sin cambio alguno, lo cual es útil cuando todas las piezas a ser tratadas son exactamente iguales.

	Fuente de energía	De energía hidráulica: En los actuadores hidráulicos fluye un líquido, comúnmente aceite. Tienen como ventaja que son pequeños comparados con la energía que proporcionan, y como desventajas que son propensos a fugas, el líquido puede incendiarse y que se requiere numeroso equipo adicional, lo cual incrementa los costos de mantenimiento del robot. Los sistemas hidráulicos están asociados a un mayor nivel de ruido.
		De energía neumática: En los actuadores neumáticos se transfiere gas bajo presión. Generalmente sólo tienen dos posiciones: retraídos y extendidos, si posibilidad de utilizar retroalimentación para usar un control proporcional. La energía neumática tiene las siguientes ventajas: está disponible en la mayoría de las áreas de manufactura, no es cara y no contamina el área de trabajo. La desventaja es que no se puede utilizar retroalimentación ni múltiples pasos.
		De energía eléctrica: Los actuadores eléctricos incluyen una fuente de poder y un motor eléctricos. La mayoría de las aplicaciones utilizan servomotores, el cual generalmente utiliza corriente directa. Las ventajas de esta fuente de energía son que no se requiere transformar la energía eléctrica en otras formas de energía como la hidráulica o neumática, no se contamina el espacio de trabajo y el nivel de ruido se mantiene bajo. La desventaja es la baja potencia que se consigue en comparación con su contraparte hidráulica.

Campos de la robótica

El campo de la robótica es muy amplio, así por ejemplo la vemos ligada a la esfera productiva, a la investigación científica, a la medicina. De acuerdo a su uso y aplicación estos tienen diversas características.

En la producción los robot se destacan por traer consigo una disminución de la mano de obra; además ayuda a una mayor calidad del producto acabado, a la rapidez de la producción. Casi siempre, en la industria, los robot se unen a otras máquinas aportando mayor eficiencia en la producción.(Ver anexo no1 Robot Industrial)

En la esfera científica, muchos de ellos son utilizados para hacer investigaciones en el campo donde el hombre se le hace difícil ir, tal vez por un medio hostil o tal vez demasiado peligroso: submarino, espacial, irradiado por centrales nucleares(Ver anexo no2 Robot para Investigaciones Científicas). Así se han diseñado dos tipos de robot de acuerdo a su misión y a su sentido de operacionalidad:

- Robot autónomo: Se le programa su misión, casi siempre con trabajos sencillos y sin necesidad de reflexionar, de comprender su entorno.

- Teleoperación o telepresencia: Esta máquina está controlada a distancia por un puesto maestro monitoreado por el operador(hombre).

En el campo de la medicina, o bien podríamos llamarle asistencia individual se destacan por la ayuda en la asistencia médica de personas paráliticas, personas con partes del cuerpo amputadas(Ver anexo no1 Robot en el campo de la medicina). La robótica entonces cubre campos como:

- Prótesis: creación de manos y piernas artificiales.
- Órtesis: estructuras rígidas motorizadas que se ponen alrededor del miembro paralizado y lo arrastran en su movimiento.
- Telétesis: destinadas a los paráliticos de los cuatro miembros(tetrapléjicos) y son robots que el afectado controla a distancia a partir de las zonas de motricidad voluntaria que haya podido conservar(por ejemplo: la lengua, los músculos de los ojos).

Desarrollo

El lenguaje siempre ha sido una vía eficaz de comunicación, las relaciones robótica–hombre también utilizan estos mecanismos para una comunicación eficaz. Entre las formas que existen de comunicación con los robots se encuentran:

- Reconocimiento de palabras separadas: actualmente este sistema es bastante primitivo y suelen depender de quien hablan. Estos sistemas pueden reconocer un conjunto de palabras concretas de un vocabulario muy limitado.
- Enseñanza y repetición: es la más comúnmente utilizada en los robots industriales. Implica el enseñar al robot todos los movimientos que necesita realizar. Normalmente la enseñanza se lleva atendiendo a los siguientes pasos:
- Dirigiendo al robot con un movimiento lento utilizando el control manual(joystick, conjunto de botones, uno para cada movimiento, o un sistema de manipulación maestro esclavo) para realizar la tarea completa y grabando los ángulos del movimiento del robot en los lugares adecuados para que vuelva a repetir el movimiento.
- Reproduciendo y repitiendo el movimiento enseñado.
- Si el movimiento enseñado es correcto, entonces se hace funcionar al robot a la velocidad correcta en el modo repetitivo.
- Lenguajes de programación de alto nivel: suministran una solución más general en la comunicación hombre–robot. Los lenguajes clásicos(FORTRAN, BASIC, PASCAL) no disponen de los comandos e instrucciones específicas que se necesitan para la programación en la robótica. Hasta ahora los lenguajes utilizados han sido diseñados para un modelo específico de manipulador, una tarea concreta, por lo que en estos momentos no existe ningún lenguaje universal.

PROGRAMACIÓN USADA EN LA ROBÓTICA

La programación que se emplea en la robótica tiene caracteres diferentes: explícito, en el que el operador es el responsable de las acciones de control y de las instrucciones adecuadas que las implementan, o estar basada en la modelación del mundo exterior, cuando se describe la tarea y el entorno y el propio sistema toma las decisiones.

La programación explícita es la más utilizada en las aplicaciones industriales y consta de dos técnicas fundamentales:

- Programación Gestual. Este tipo de programación, exige el empleo del manipulador en la fase de enseñanza, o sea, trabaja "on–line".

- Programación Textual. En esta labor no participa la máquina (off-line). Las trayectorias del manipulador se calculan matemáticamente con gran precisión y se evita el posicionamiento a ojo.

PROGRAMACIÓN GESTUAL o directa

Es en este tipo de programación donde el propio brazo interviene en el trazado del camino y en las acciones a desarrollar en la tarea de la aplicación; lo que determina la programación "on-line". Esta está dividida en dos clases:

- Programación por aprendizaje directo: El punto final del brazo se traslada con ayuda de un dispositivo especial colocado en su muñeca, o utilizando un brazo maestro o maniquí, sobre el que se efectúan los desplazamientos que, tras ser memorizados, serán repetidos por el manipulador. Esta programación tiene pocas posibilidades de edición ya que para generar una trayectoria continua, es preciso almacenar o definir una gran cantidad de puntos, cuya reducción origina discontinuidades.
- Programación mediante un dispositivo de enseñanza: Consiste en determinar las acciones y movimientos del brazo manipulador, a través de un elemento especial para este cometido. En este caso, las operaciones ordenadas se sincronizan para conformar el programa de trabajo. Los dispositivos de enseñanza modernos permiten generar funciones auxiliares, además del control de los movimientos:
 - Selección de velocidades
 - Generación de retardos
 - Señalización del estado de los sensores
 - Borrado y modificación de los puntos de trabajo
 - Funciones especiales

Esta programación tiene como característica común que el usuario no necesita conocer ningún lenguaje de programación, simplemente debe habituarse al empleo de los elementos que constituyen el dispositivo de enseñanza. De esta forma, se pueden editar programas, aunque como es lógico, muy simples. Los lenguajes de programación gestual, además de necesitar al propio robot en la confección del programa, carecen de adaptabilidad en tiempo real con el entorno y no pueden tratar, con facilidad, interacciones de emergencia.

PROGRAMACIÓN TEXTUAL

El programa queda constituido por un texto de instrucciones o sentencias, cuya confección no requiere de la intervención del robot; es decir, se efectúan "off-line". Con este tipo de programación, el operador no define, prácticamente, las acciones del brazo manipulado, sino que se calculan, en el programa, mediante el empleo de las instrucciones textuales adecuadas.

En una aplicación tal como el ensamblaje de piezas, en la que se requiere una gran precisión, los posicionamientos seleccionados mediante la programación gestual no son suficientes, debiendo ser sustituidos por cálculos más perfectos y por una comunicación con el entorno que rodea al sistema. En esta la posibilidad de edición es total. El robot debe intervenir, sólo, en la puesta a punto final.

Según las características del lenguaje, pueden confeccionarse programas de trabajo complejos, con inclusión de saltos condicionales, empleo de bases de datos, posibilidad de creación de módulos operativos intercambiables, capacidad de adaptación a las condiciones del mundo exterior, etc.

Esta programación textual está dividida en dos grandes grupos de diferencias marcadas:

- Programación textual explícita.
- Programación textual especificativa.

Programación textual explícita

En la programación textual explícita, el programa consta de una secuencia de órdenes o instrucciones concretas, que van definiendo con rigor las operaciones necesarias para llevar a cabo la aplicación. Se puede decir que la programación explícita engloba a los lenguajes que definen los movimientos punto por punto, similares a los de la programación gestual, pero bajo la forma de un lenguaje formal. Con este tipo de programación, la labor del tratamiento de las situaciones anormales, colisiones, etc., queda a cargo del programador.

Dentro de la programación explícita, hay dos niveles:

- Nivel de movimiento elemental que comprende los lenguajes dirigidos a controlar los movimientos del brazo manipulador. Existen dos tipos:
 - Articular, cuando el lenguaje se dirige al control de los movimientos de las diversas articulaciones del brazo.
 - Cartesiano, cuando el lenguaje define los movimientos relacionados con el sistema de manufactura, es decir, los del punto final del trabajo (TCP).

Los lenguajes del tipo cartesiano utilizan transformaciones homogéneas, lo que hace que se independice a la programación del modelo particular del robot, puesto que un programa confeccionado para uno, en coordenadas cartesianas, puede utilizarse en otro, con diferentes coordenadas, mediante el sistema de transformación correspondiente.

Por el contrario, los lenguajes del tipo articular indican los incrementos angulares de las articulaciones. Aunque esta acción es bastante simple para motores de paso a paso y corriente continua, al no tener una referencia general de la posición de las articulaciones con relación al entorno, es difícil relacionar al sistema con piezas móviles, obstáculos, cámaras de TV, etc.

- Nivel estructurado, el que intenta introducir relaciones entre el objeto y el sistema del robot, para que los lenguajes se desarrollen sobre una estructura formal.

Se puede decir que los lenguajes correspondientes a este tipo de programación adoptan la filosofía del PASCAL. Describen objetos y transformaciones con objetos, disponiendo, muchos de ellos, de una estructura de datos arborescente.

El uso de lenguajes con programación explícita estructurada aumenta la comprensión del programa, reduce el tiempo de edición y simplifica las acciones encaminadas a la consecución de tareas determinadas.

En los lenguajes estructurados, es típico el empleo de las transformaciones de coordenadas, que exigen un cierto nivel de conocimientos. Por este motivo dichos lenguajes no son populares hoy en día.

Programación textual especificativa

La programación textual explícita es una programación del tipo no procesal, en la que el usuario

describe las especificaciones de los productos mediante una modelización, al igual que las tareas que hay que realizar sobre ellos.

El sistema informático para la programación textual especificativa ha de disponer del modelo del universo (actualmente, los modelos del universo son del tipo geométrico, no físico), o mundo donde se encuentra el robot. Este modelo será, normalmente, una base de datos más o menos compleja, según la clase de aplicación, pero que requiere, siempre, computadoras potentes para el procesamiento de una abundante información. El trabajo de la programación consistirá, simplemente, en la descripción de las tareas a realizar, lo que supone poder llevar a cabo trabajos complicados.

lenguajes de PROGRAMACIÓN

A continuación se realiza una descripción de los lenguajes de programación más usados en la robótica.

GESTUAL PUNTO A PUNTO

Se aplican con el robot "in situ", recordando a las normas de funcionamiento de un magnetófono doméstico, ya que disponen de unas instrucciones similares: PLAY (reproducir), RECORD (grabar), FF (adelantar), FR (atrasar), PAUSE, STOP, etc. Además, puede disponer de instrucciones auxiliares, como INSERT (insertar un punto o una operación de trabajo) y DELETE (borrar). Este manipulador en línea funciona como un digitalizador de posiciones.

Los lenguajes más conocidos en programación gestual punto a punto son el FUNKY, creado por IBM para uno de sus robots, y el T3, original de CINCINNATI MILACROM para su robot T3. Los movimientos pueden tener lugar en sistemas de coordenadas cartesianas, cilíndricas o de unión, siendo posible insertar y borrar las instrucciones que se desee. Es posible, también, implementar funciones relacionadas con sensores externos, así como revisar el programa paso a paso, hacia delante y hacia atrás.

En el lenguaje FUNKY se usa un mando del tipo "joystick", que dispone de un comando especial para centrar a la pinza sobre el objeto para el control de los movimientos, mientras que el T3 dispone de un dispositivo de enseñanza ("teach pendant").

El procesador usado en T3 es el AMD 2900 ("bit slice"), mientras que en el FUNKY está constituido por el IBM SYSTEM-7.

A NIVEL DE MOVIMIENTOS ELEMENTALES.

Los movimientos de punto a punto también se expresan en forma de lenguaje:

- ◆ **ANORAD**
- ◆ **EMILY**
- ◆ **RCL**
- ◆ **RPL**
- ◆ **SIGLA**
- ◆ **VAL**
- ◆ **MAL**

Todos ellos mantienen el énfasis en los movimientos primitivos, ya sea en coordenadas articulares, o cartesianas. En comparación, tienen, como ventajas destacables, los saltos condicionales y a subrutina, además de un aumento de las operaciones con sensores, aunque siguen manteniendo pocas posibilidades de programación "off-line".

Estos lenguajes son, por lo general, del tipo intérprete, con excepción del RPL, que tiene un compilador. La mayoría dispone de comandos de tratamiento a sensores básicos: tacto, fuerza, movimiento, proximidad y presencia. El RPL dispone de un sistema complejo de visión, capaz de seleccionar una pintura y reconocer objetos presentes en su base de datos.

Los lenguajes EMILY y SIGLA son transportables y admiten el proceso en paralelo simple.

Otros datos interesantes de este grupo de lenguajes son los siguientes:

ANORAD

Se trata de una transformación de un lenguaje de control numérico de la casa ANORAD CORPORATION, utilizado para robot ANOMATIC. Utiliza, como procesador, al microprocesador 68000 de Motorola de 16/32 bits.

VAL

Fue diseñado por UNIMATION INC para sus robots UNIMATE y PUMA. (FIG. 1) Emplea, como CPU, un LSI-II, que se comunica con procesadores individuales que regulan el servocontrol de cada articulación. Las instrucciones, en idioma inglés, son sencillas e intuitivas, como se puede apreciar por el programa siguiente:

LISPT

PROGRAM PICKUP

1. APRO PART, 25.0

2. MOVES PART

3. CLOSE, 0.0.0

4. APRO PART, -50.0

5. APRO DROP, 100.0

6. MOVES DROP

7. OPEN, 0.0.0

8. APRO DROP, -100.0

END

RPL

Dotado con un LSI-II como procesador central, y aplicado a los robots PUMA, ha sido diseñado por SRI INTERNATIONAL.

EMILY

Es un lenguaje creado por IBM para el control de uno de sus robots. Usa el procesador IBM 370/145

SYSTEM 7 y está escrito en Ensamblador.

SIGLA

Desarrollado por OLIVETTI para su robot SUPER SIGMA, emplea un mini-ordenador con 8 K de memoria. Escrito en Ensamblador, es del tipo intérprete.

MAL

Se ha creado en el Politécnico de Milán para el robot SIGMA, con un Mini-multiprocesador. Es un lenguaje del tipo intérprete, escrito en FORTRAN.

RCL

Aplicado al robot PACS y desarrollado por RPI, emplea, como CPU, un PDP 11/03. Es del tipo intérprete y está escrito en Ensamblador.

ESTRUCTURADOS DE PROGRAMACIÓN EXPLÍCITA

Teniendo en cuenta las importantísimas características que presenta este tipo de programación, merecen destacarse los siguientes lenguajes:

- ◆ **AL**
- ◆ **HELP**
- ◆ **MAPLE**
- ◆ **PAL**
- ◆ **MCL**
- ◆ **MAL EXTENDIDO**

Con excepción de HELP, todos los lenguajes de este grupo están provistos de estructuras de datos del tipo complejo. Así, el AL utiliza vectores, posiciones y transformaciones; el PAL usa, fundamentalmente, transformaciones y el MAPLE permite la definición de puntos, líneas, planos y posiciones. Sólo el PAL, y el HELP carecen de capacidad de adaptación sensorial. Los lenguajes AL, MAPLE y MCL, tienen comandos para el control de la sensibilidad del tacto de los dedos (fuerza, movimiento, proximidad, etc.). Además, el MCL posee comandos de visión para identificar e inspeccionar objetos.

A continuación, se exponen las características más representativas de los lenguajes dedicados a la programación estructurada.

AL

Trata de proporcionar definiciones acerca de los movimientos relacionados con los elementos sobre los que el brazo trabaja. Fue diseñado por el laboratorio de Inteligencia Artificial de la Universidad de Stanford, con estructuras de bloques y de control similares al ALGOL, lenguaje en el que se escribió. Está dedicado al manipulador de Stanford, utilizando como procesadores centrales, a un PDP 11/45 y un PDP KL-10.

HELP

Creado por GENERAL ELECTRIC para su robot ALLEGRO y escrito en PASCAL/FORTRAN, permite el movimiento simultáneo de varios brazos. Dispone, asimismo, de un conjunto especial de subrutinas para la ejecución de cualquier tarea. Utilizando como CPU, a un PDP 11.

MAPLE

Escrito, como intérprete, en lenguaje PL-1, por IBM para el robot de la misma empresa, tiene capacidad para soportar informaciones de sensores externos. Utiliza, como CPU a un IBM 370/145 SYSTEM 7.

PAL

Desarrollado por la Universidad de Purdue para el manipulador de Stanford, es un intérprete escrito en FORTRAN y Ensamblador, capaz de aceptar sensores de fuerza y de visión. Cada una de sus instrucciones, para mover el brazo del robot en coordenadas cartesianas, es procesada para que satisfaga la ecuación del procesamiento. Como CPU, usan un PDP 11/70.

MCL

Lo creó la compañía MC DONALL DOUGLAS, como ampliación de su lenguaje de control numérico APT. Es un lenguaje compilable que se puede considerar apto para la programación de robots "off-line".

MAL EXTENDIDO

Procede del Politécnico de Milán, al igual que el MAL, al que incorpora elementos de programación estructurada que lo potencian notablemente. Se aplica, también, al robot SIGMA.

ESPECIFICATIVA A NIVEL OBJETO.

En este grupo se encuentran tres lenguajes interesantes:

- ◆ **RAPT**
- ◆ **AUTOPASS**
- ◆ **LAMA**

RAPT

Su filosofía se basa en definir una serie de planos, cilindros y esferas, que dan lugar a otros cuerpos derivados. Para modelar a un cuerpo, se confecciona una biblioteca con sus rasgos más representativos. Seguidamente, se define los movimientos que ligan a los cuerpos a ensamblar (alineación, encajar cilindros, etc.).

Así, si se desea definir un cuerpo C1, se comienza definiendo sus puntos más importantes, por ejemplo:

$$P1 = \langle x, 0, 0 \rangle$$

$$P2 = \langle 0, y, 0 \rangle$$

$$P3 = \langle x/2, y, 0 \rangle$$

$$P4 = \langle 0, 0, z \rangle$$

Si, en el cuerpo, existen círculos de interés, se especifican seguidamente:

$$C1 = \text{CIRCLE}/P2, R;$$

C2 = CIRCLE/P4, R;

A continuación, se determinan sus aristas:

L1 = L/P1, P2;

L2 = L/P3, P4;

Si, análogamente al cuerpo C1, se define otro, como el C2, una acción entre ambos podría consistir en colocar la cara inferior de C1 alineada con la superior de C2. Esto se escribiría.

AGAINST / BOT / OF C1, TOP / OF C2;

El lenguaje RAPT fue creado en la Universidad de Edimburgo, departamento de Inteligencia Artificial; está orientado, en especial, al ensamblaje de piezas. Destinado al robot FREDY, utiliza, como procesador central, a un PDP 10. Es un intérprete y está escrito en lenguaje APT.

AUTOPASS

Creado por IBM para el ensamblaje de piezas; utiliza instrucciones, muy comunes, en el idioma inglés. Precisa de un ordenador de varios Megabytes de capacidad de memoria y, además de indicar, como el RAPT, puntos específicos, prevé, también, colisiones y genera acciones a partir de las situaciones reales.

Un pequeño ejemplo, que puede proporcionar una idea de la facilidad de relacionar objetos, es el programa siguiente, que coloca la parte inferior del cuerpo C1 alineada con la parte superior del cuerpo C2. Asimismo, alinea los orificios A1 y A2 de C1, con los correspondientes de C2.

PLACE C1

SUCH THAT C1 BOT CONTACTS C2TOP

AND B1 A1 IS ALIGNED WITH C2A1

AND B1 A2 IS ALIGNED WITH C2A2

El AUTOPASS realiza todos sus cálculos sobre una base de datos, que define a los objetos como poliedros de un máximo de 20,000 caras. Está escrito en PL/1 y es intérprete y compilable.

LAMA

Procede del laboratorio de Inteligencia Artificial del MIT, para el robot SILVER, orientándose hacia el ajuste de conjuntos mecánicos. Aporta más inteligencia que el AUTOPASS y permite una buena adaptación al entorno. La operatividad del LAMA se basa en tres funciones principales:

- ◆ Creación de la función de trabajo. Operación inteligente.
- ◆ Generación de la función de manipulación.
- ◆ Interpretación y desarrollo, de una forma interactiva, de una estrategia de realimentación para la adaptación al entorno de trabajo.

EN FUNCIÓN DE LOS OBJETIVOS.

La filosofía de estos lenguajes consiste en definir la situación final del producto a fabricar, a partir de

la cual se generan los planes de acción tendentes a conseguirla, obteniéndose, finalmente, el programa de trabajo. Estos lenguajes, de tipo natural, suponiendo una potenciación extraordinaria de la Inteligencia Artificial, para descargar al usuario de las labores de programación. Prevén, incluso, la comunicación hombre-máquina a través de la voz.

Los lenguajes más conocidos de este grupo son:

◆ **STRIPS**

◆ **HILAIRE**

STRIPS

Fue diseñado, en la Universidad de Stanford, para el robot móvil SHAKEY. Se basa en un modelo del universo ligado a un conjunto de planteamientos aritmético-lógicos que se encargan de obtener las subrutinas que conforman el programa final. Es intérprete y compilable, utilizando, como procesadores, a un PDP-10 y un PDP-15.

HILAIRE

Procedente del laboratorio de Automática Y Análisis de Sistemas (LAAS) de Toulouse, está escrito en lenguaje LISP. Es uno de los lenguajes naturales más interesantes, por sus posibilidades de ampliación e investigación.

Conclusiones

Es mucho lo que se ha hecho en el área de la programación para la robótica; sin embargo aún no existe un lenguaje ideal para la programación de los robots. Son muchos los lenguajes creados hasta ahora, en parte las causas principales de esta amplia gama de lenguajes inadecuados o poco efectivos son:

- Cada lenguaje se ha diseñado tomando como base un robot en específico del mercado, lo que anula su universalidad y la posibilidad de emplearlo en modelos diferentes.
- Los lenguajes, en muchos casos, se dirigen hacia aplicaciones diferentes, lo que limita grandemente su utilización para la programación de otras tareas.

Hoy en el mundo existe un interés general para lograr un sistema de percepción del entorno cada vez más avanzado. Para esto se hace necesaria la ampliación de la Inteligencia Artificial, que interviene en la valoración del espacio exterior o entorno y determina los planes de acción alternativos o lo que es lo mismo la respuesta a la interacción con ese medio.

CARACTERÍSTICAS DE UN LENGUAJE IDEAL PARA LA ROBÓTICA

Las seis características básicas de un lenguaje ideal, expuestas por Pratt, son:

1. Claridad y sencillez.
2. Claridad de la estructura del programa.
3. Sencillez de aplicación.
4. Facilidad de ampliación.
5. Facilidad de corrección y mantenimiento.

6. Eficacia.

Estas características son insuficientes para la creación de un lenguaje "universal" de programación en la robótica, por lo que es preciso añadir las siguientes:

- Transportabilidad sobre cualquier equipo mecánico o informático.
- Adaptabilidad a sensores (tacto, visión, etc.).
- Posibilidad de descripción de todo tipo de herramientas acoplables al manipulador.
- Interacción con otros sistemas.

En el aspecto de claridad y sencillez, la programación gestual es la más eficaz, pero impide la confección de programas propiamente dichos. Los lenguajes a nivel de movimientos elementales, como el VAL, disponen de bastantes comandos para definir acciones muy parecidas que fueron surgiendo según las necesidades y que, en gran medida, oscurecen su comprensión y conocimiento. Aunque, inicialmente, las técnicas de programación estructurada son más difíciles de dominar, facilitan, extraordinariamente, la comprensión y corrección de los programas.

Respecto a la sencillez de aplicación, hay algunos lenguajes (como el MCL) dedicados a las máquinas herramienta (APT), que pueden ser valorados, positivamente, por los usuarios conocedores de este campo. El PAL, estructurado sobre la matemática matricial, sólo es adecuado para quienes están familiarizados con el empleo de este tipo de transformaciones.

Uno de los lenguajes más fáciles de utilizar es el AUTOPASS, que posee un juego de comandos con una sintaxis similar a la del inglés corriente.

Es imprescindible que los lenguajes para los robots sean fácilmente ampliables, por lo que se les debe dotar de una estructura modular, con inclusión de subrutinas definidas por el mismo usuario.

La adaptabilidad a sensores externos implica la posibilidad de una toma de decisiones, algo muy interesante en las labores de ensamblaje. Esta facultad precisa de un modelo dinámico del entorno, así como de una buena dosis de Inteligencia Artificial, como es el caso del AUTOPASS.

Aunque los intérpretes son más lentos que los compiladores, a la hora de la ejecución de un programa, resultan más adecuados para las aplicaciones de la robótica. Las razones son las siguientes:

- El intérprete ejecuta el código como lo encuentra, mientras que el compilador recorre el programa varias veces, antes de generar el código ejecutable.
- Los intérpretes permiten una ejecución parcial del programa.
- La modificación de alguna instrucción es más rápida con intérpretes, ya que un cambio en una de ellas no supone la compilación de las demás.

Finalmente, el camino para la superación de los problemas propios de los lenguajes actuales ha de pesar, necesariamente, por la potenciación de los modelos dinámicos del entorno que rodea al robot, acompañado de un aumento sustancial de la Inteligencia Artificial.

Anexos

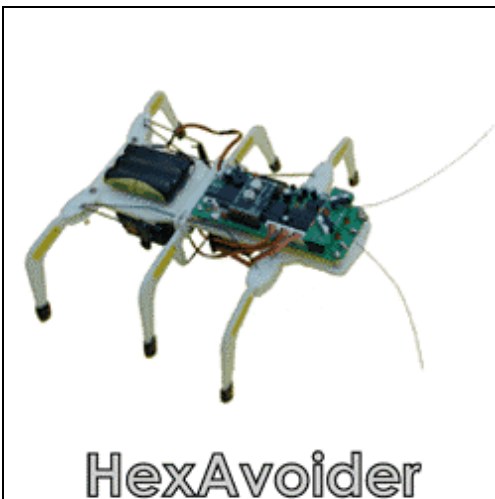
Ejemplos de robots

Artículo sobre robot antibombas.



Bomb Buster: The *Mini-Andros* is used by bomb squads across the country to locate and dispose of bombs. About three feet long, the *Mini-Andros* looks something like a small armored tank with eight wheels on four "legs" that extend for climbing stairs. Its movable arm can lift objects weighing up to 15 pounds and place them in bomb-proof boxes. Detachable accessories let the *Mini-Andros* to break windows, see in the dark, and to defuse or detonate bombs directly, either by blasting them with water, firing at them with a shotgun, or placing other smaller bombs nearby.

Artículo sobre microrobot



The HexAvoider is a programmable mobile robot designed to move independently and avoid obstacles by either moving to the left or right. HexAvoider uses infrared emitters and receivers to "sense" its environment. It also has two antennae or "feelers". These are also used to sense obstacles.

HexAvoider is programmed using C language. The user can program on his PC and download the program to the robot using FLASH memory. The user has the option of including a RF module in HexAvoider's construction. With this, the operator can directly control HexAvoider's movements from his PC.

HexAvoider comes in kit form. The front and rear legs provide locomotion while the middle pair steers the robot. The body is made of high quality, durable plastic and is quite rugged.

Descripción de un robot en funcionamiento

ANGULO, J. M.: Curso de robótica. Editorial Revolucionaria.

COIFFET, P.: Elementos de robótica. Editorial Revolucionaria.

DUEÑAS, F. A.: La Robótica. www.monografias.com

Universidad La Salle – Ing. Cibernética.

CANDELAS, S.(UNAM) : Lenguajes de programación de los Robots www.monografias.com

<http://www.thetech.org/robotics/universal/page02.html>

<http://www.microrobotna.com/discription/hexapod.htm>

COIFFET, P.: Elementos de robótica. Editorial Revolucionaria.

22

Programación en robótica

Aprendizaje directo

Mediante dispositivos de enseñanza (teclado, joystic)

Gestual

Textual

Explícita (movimiento)

Estructurada

Movimientos elementales

Articular

Cartesiana

Especificativa (modelo)

Modelo orientado a objetos

Modelo orientado a objetivos o tareas

Tareas a realizar

Captadores exteroceptivos

Entorno

Interacción

Información propioceptiva

Control

Lenguajes

Información exteroceptiva

Ordenador

Modelo del robot

Modelo del entorno

Descripción de las tareas

Algoritmos de control

Captadores propioceptivos

Sistema mecánico articulado

Transmisiones

Accionadores