

UNIVERSIDAD NACIONAL DE INGENIERIA
Facultad de Ingeniería Eléctrica y Electrónica



**“DISEÑO Y CONSTRUCCION DE UN ROBOT TIPO
SCARA DE CUATRO GRADOS DE LIBERTAD”**

TESIS

Para optar el Título profesional de:

Ingeniero Electrónico

Presentado por:

Machuca Mines José Ambrosio

Promoción 1990-I

***LIMA - PERU
1999***

**DISEÑO Y CONSTRUCCION DE UN ROBOT TIPO
SCARA DE CUATRO GRADOS DE LIBERTAD**

Sumario

En el presente trabajo se analizan los conceptos fundamentales de Robótica y se describe el diseño, la implementación y el control del prototipo de un robot de cuatro grados de libertad llamado *ROB-DEX*, caracterizado por su total originalidad, el cual está constituido por lo siguiente:

- Un sistema electrónico implementado a base de un microprocesador y una arquitectura programable distribuido en módulos para procesar las señales eléctricas de entrada/salida del sistema computador-robot.
- Un sistema informático elaborado en base de lenguajes de programación general para procesar los algoritmos matemáticos, de control y simulación gráfica del funcionamiento físico del robot en tiempo real.
- Un sistema automático desarrollado en base a técnica avanzadas de sistemas de control en Robótica, para controlar la posición, orientación, trayectoria, movimiento y velocidad del robot, respecto de un sistema de referencia.
- Un sistema mecánico construido en base a elementos metálicos, unidos por articulaciones accionadas por motores a través de un sistema de engranajes y dotado de sensores, para registrar la posición y velocidad de cada articulación.

El *ROB-DEX* puede ser considerado como una plataforma de aplicación de los conocimientos de Robótica siendo éste, flexible y reprogramable, potencialmente se puede aplicar en la industria productiva dentro del margen de su ejecución de trabajo.

INDICE

| | Página |
|--|--------|
| PROLOGO | 01 |
| CAPITULO I | |
| GENERALIDADES | 04 |
| 1.1 Introducción. | 04 |
| 1.2 Definición de robot. | 06 |
| 1.3 Clasificación de los robots. | 08 |
| 1.4 Especificaciones técnicas de los robots. | 21 |
| 1.5 Componentes principales de un robot. | 26 |
| 1.5.1 El manipulador. | 26 |
| 1.5.2 Actuadores. | 27 |
| 1.5.3 Sensores. | 30 |
| 1.5.4 Elementos de transmisión. | 31 |
| 1.5.5 El controlador. | 33 |
| 1.5.6 Unidad de conversión de energía. | 33 |
| 1.6 El <i>ROB-DEX</i> . | 34 |
| CAPITULO II | |
| CINEMATICA DEL ROBOT | 37 |
| 2.1 Introducción. | 37 |

| | | |
|--|---|-----------|
| 2.2 | Estructura cinemática | 39 |
| 2.3 | Representación de Denavit-Hartenberg. | 41 |
| 2.3.1 | Algoritmo para la representación de D-H. | 48 |
| 2.4 | Especificaciones del efector final. | 50 |
| 2.5 | Cinemática directa. | 55 |
| 2.5.1 | Cinemática directa del <i>ROB-DEX</i> . | 57 |
| 2.6 | Cinemática inversa. | 62 |
| 2.6.1 | Cinemática inversa del <i>ROB-DEX</i> . | 65 |
| 2.7 | Singularidades. | 66 |
| CAPITULO III | | |
| PLANIFICACION Y CONTROL DE TRAYECTORIAS DEL ROBOT | | 68 |
| 3.1 | Introducción. | 68 |
| 3.2 | Análisis del espacio de trabajo. | 69 |
| 3.3 | Planificación de la trayectoria. | 72 |
| 3.4 | Clasificación de las trayectorias. | 73 |
| 3.5 | Generación de trayectorias articulares. | 76 |
| 3.5.1 | Interpolación polinómica lineal. | 76 |
| 3.5.2 | Interpolación polinómica cúbica. | 79 |
| 3.5.3 | Interpolación polinómica de grado superior. | 84 |
| 3.6 | Generación de trayectorias cartesianas. | 90 |
| 3.6.1 | Interpolación lineal del vector de variables cartesianas. | 92 |
| 3.6.2 | Interpolación lineal mediante transformación homogénea. | 94 |
| 3.6.3 | Interpolación lineal de una rotación y traslación. | 101 |
| 3.7 | Consideraciones finales. | 103 |

CAPITULO IV

| | |
|--|-----|
| DINAMICA DEL ROBOT | 105 |
| 4.1 Introducción. | 105 |
| 4.2 Modelamiento dinámico. | 106 |
| 4.3 Formulación de Lagrange. | 106 |
| 4.3.1 Energía cinética. | 108 |
| 4.3.2 Energía potencial | 116 |
| 4.3.3 Fuerzas generalizadas. | 117 |
| 4.3.4 Modelo dinámico de Lagrange - Euler. | 118 |
| 4.4 Dinámica del <i>ROB-DEX</i> . | 123 |
| 4.5 Propiedades del modelo dinámico. | 137 |

CAPITULO V

| | |
|--|-----|
| SISTEMA DE CONTROL DEL ROBOT | 142 |
| 5.1 Introducción. | 142 |
| 5.2 El problema de control. | 143 |
| 5.3 Ecuaciones de estado. | 146 |
| 5.4 Métodos de control. | 147 |
| 5.4.1 Control posicional PID de simple articulación. | 149 |
| 5.4.2 Control PD con compensación de la gravedad. | 165 |
| 5.4.3 Control mediante el método del torque calculado. | 170 |
| 5.5 Control del <i>ROB-DEX</i> . | 177 |

CAPITULO VI

| | |
|---------------------------------|-----|
| PROGRAMACION DE UN ROBOT | 186 |
| 6.1 Introducción. | 186 |

| | | |
|--|--|-----|
| 6.2 | Programación no textual. | 188 |
| 6.3 | Programación a nivel de robot | 189 |
| 6.3.1 | Ventajas e inconvenientes. | 191 |
| 6.3.2 | Consideraciones respecto al diseño informático. | 191 |
| 6.4 | Programación a nivel tarea. | 192 |
| 6.4.1 | Consideraciones acerca de su campo de aplicación. | 193 |
| 6.4.2 | Sistemas de programación textuales a nivel tarea, textuales. | 195 |
| 6.4.3 | Sistemas de programación basados en el modelado geométrico de sólidos. | 197 |
| 6.5 | Lenguajes naturales y programación a nivel objetivo. | 198 |
| 6.5.1 | Lenguajes naturales. | 198 |
| 6.5.2 | Programación a nivel objetivo. | 199 |
| 6.6 | Presente y requisitos futuros de la programación de robots. | 199 |
| 6.6.1 | Los sistemas de programación y el robot. | 200 |
| 6.6.2 | Relación del robot con el entorno de fabricación. | 201 |
| 6.6.3 | Los sistemas de programación y el usuario. | 201 |
| 6.7 | Programación del <i>ROB-DEX</i> . | 203 |
| CAPITULO VII | | |
| ARQUITECTURA ELECTRONICA DEL <i>ROB-DEX</i> | | 208 |
| 7.1 | Introducción. | 208 |
| 7.2 | Sistema electrónico. | 209 |
| 7.2.1 | Unidad de interfaz con el computador. | 211 |
| 7.2.2 | Unidad de control y regulación del robot. | 211 |
| 7.2.3 | Unidad de impulsión de las articulaciones. | 215 |
| 7.2.4 | Unidad de sensado de articulaciones. | 219 |

| | | |
|--|---|-----|
| 7.2.5 | Unidad de expansión del controlador | 224 |
| CAPITULO VIII | | |
| APLICACIONES DEL ROB-DEX | | 226 |
| 8.1 | Aplicaciones para la enseñanza de Robótica. | 226 |
| 8.2 | Aplicaciones industriales. | 226 |
| CAPITULO IX | | |
| COSTOS DEL PROYECTO ROB-DEX | | 228 |
| 9.1 | Introducción. | 228 |
| 9.2 | Costos de materiales. | 228 |
| 9.3 | Costos de construcción. | 232 |
| 9.4 | Costos de diseño de Ingeniería. | 232 |
| 9.5 | Costos de prueba y puesta a punto. | 232 |
| CONCLUSIONES | | 233 |
| ANEXO A | | |
| MATRICES DE ROTACION Y TRASLACION | | 236 |
| A.1 | Matrices de rotación. | 237 |
| A.2 | Matriz de rotación compuesta. | 242 |
| A.2.1 | Interpretación geométrica de las matrices de rotación. | 243 |
| A.3 | Matriz de rotación respecto de un eje arbitrario. | 245 |
| A.4 | Matriz de rotación con representación de los ángulos de Euler. | 247 |
| A.5 | Coordenadas homogéneas y matriz de transformación. | 249 |
| A.6 | Matriz de transformación homogénea compuesta. | 251 |
| A.6.1 | Interpretación geométrica de la matriz de transformación homogénea. | 253 |
| A.7 | Algunas matrices de transformación homogéneas compuestas. | 255 |

ANEXO B**CALCULO DE UNA TRAYECTORIA DE ORDEN SUPERIOR** 260

B.1 Cálculo de una trayectoria de articulación 4-3-4. 261

ANEXO C**FUNDAMENTOS PARA LA APLICACIÓN DE LAGRANGE** 269

C.1 Fuerzas generalizadas. 270

C.2 Actuadores. 270

C.3 Fuerzas de fricción. 271

C.4 Tensor de inercia del eslabón. 274

ANEXO D**FUNDAMENTOS DE ESTABILIDAD DE SISTEMAS NO LINEALES** 277

D.1 Soluciones constantes. 278

D.2 Estabilidad asintótica. 281

D.3 Primer método de Liapunov. 282

D.4 Segundo método de Liapunov. 284

BIBLIOGRAFIA 287

PROLOGO

Este trabajo fue desarrollado con el propósito de proporcionar los principios básicos de diseño, análisis, implementación y control del sistema robot.

El trabajo consiste en el diseño, realización y control de un prototipo de robot de cuatro grados de libertad que se le ha denominado *ROB-DEX*.

Los primeros estudios de Robótica los realicé de forma autodidacta, leyendo revistas y artículos desde que estaba cursando mis estudios en esta Universidad. Posteriormente tuve acceso a libros de esta temática; de manera que iba entendiendo cada vez más la "filosofía" de un robot. Al finalizar mi carrera, me dedique casi exclusivamente a escudriñar libros que trataban los conceptos básicos de Robótica, hasta comprender cada vez más robots más complejos.

En el año 1993 conocí al Dr. Oscar Kurisaki, graduado en Cibernética y Automática en Alemania, él es quien madura mis conocimientos de Robótica. En ese mismo tiempo retorna a la UNI el Dr. Luis Herrera Bendezú, graduado en EE.UU, quién también contribuye a solidificar mis conocimientos de Robótica.

En el año 1994 me dispuse a implementar al *ROB-DEX* como una materialización de los conceptos adquiridos con el propósito de presentarlo en la feria de TECNOTRON. Obviamente la implementación fue muy simple y no tenía ningún tipo de control; es decir no realizaba ningún trabajo concreto, tan sólo movimientos aleatorios. Pese a ello, tuvo bastante aceptación y creó expectativas lo que me motivó

bastante en seguir mejorándolo. Al finalizar ese mismo año se realiza por primera vez en el Perú el I CONGRESO INTERNACIONAL DE INGENIERIA ELECTRONICA (INTERCON I) organizado por la rama estudiantil de la IEEE en la Universidad Ricardo Palma, motivo por el cual participé en el “Concurso de Trabajos de Investigación” convocado por dicho congreso en representación de la UNIVERSIDAD NACIONAL DE INGENIERIA llegando a obtener el primer lugar, lo que fue un aliciente para seguir investigando más sobre Robótica y avanzando en el mejoramiento del *ROB-DEX*. En ese entonces ya se logró controlar la trayectoria y la velocidad del efector final por lo que ya realizaba tareas sencillas, como por ejemplo escribir UNI en un plano de dos dimensiones. En ese mismo año La Universidad San Antonio de Abad del Cuzco realiza el I CONGRESO NACIONAL DE ESTUDIANTES DE INGENIERIA MECANICA ELECTRICA Y RAMAS A FINES (I CONEIMERA), logrando obtener con el *ROB-DEX* el *Premio de excelencia al mejor trabajo presentado*.

En el año de 1995 publiqué un artículo del *ROB-DEX* en el XII CONIMERA realizado en el Perú y otro artículo en el XI CONGRESO CHILENO DE INGENIERIA ELECTRICA organizado por la Universidad de Magallanes en la ciudad de Punta Arenas en Chile.

En la actualidad el *ROB-DEX* se encuentra implementado casi en su totalidad, el cual ha sido expuesto en muchas ocasiones, por ejemplo, con motivo de la semana de la FIEE, EXPOUNI y otros.

El *ROB-DEX* ha sido valorado por la originalidad de su diseño, construcción, por los elementos usados, por su estructura electrónica y por la programación de su control.

Estoy en deuda con una serie de personas que, directa o indirectamente, me ayudaron, me recomendaron, me sugirieron, me animaron en el arduo trabajo de la investigación científica, que sería muy numerosa la lista de quienes contribuyeron de alguna manera con este trabajo. En una forma muy especial mis agradecimientos a todos mis profesores que me dieron sus enseñanzas.

Deseo agradecer en forma especial al Dr Arturo Rojas por la revisión detallada de esta tesis y por las sugerencias impartidas para la culminación de este trabajo.

José A. Machuca Mines.

CAPITULO I

GENERALIDADES

1.1 Introducción

La Robótica es un campo multidisciplinario complejo en el cual intervienen diversas especialidades de Ingeniería, tales como Ing. Electrónica, Ing. Mecánica, Ing. de Control, Ing. Informática, Procesamiento de señales, Visión Artificial, Sistemas de Manufactura, etc.

La Robótica es un área moderna, científica y tecnológica que se viene desarrollando a velocidades vertiginosas en los países desarrollados y en aquellos aún en vías de desarrollo, sobre todo en estas cuatro últimas décadas, tanto en la investigación como en sus diferentes áreas de aplicación: industriales, militares, espaciales, educacionales, etc.

La Robótica se relaciona estrechamente con la Automatización Industrial, por lo que fundamentalmente debe estar apoyada por la industria moderna interesada en aminorar los costos y el tiempo de productividad, aumentar la flexibilidad y calidad de la producción masiva.

Los robots son los elementos más representativos de los actuales sistemas modernos de producción, con un elevado grado de flexibilidad y adaptabilidad en sus aplicaciones. El crecimiento exponencial de la línea mundial de robots en estos últimos años y su creciente diversificación ha dado lugar a la aparición de la

ROBOTICA INDUSTRIAL, como una tecnología factible del futuro, cuya filosofía de su aplicación e investigación esencialmente se basa en:

- Que un robot no puede actuar en contra del ser humano o mediante la inacción, permitir que éste sufra daños.
- Disminuir el trabajo monótono y agotador del hombre.
- Que los robots realicen tareas en ámbitos inhóspitos al ser humano.
- Que el trabajo humano se oriente a las tareas creativas en vez de las productivas.
- Elevar la producción masiva y el rendimiento en la industria.
- La disminución de los costos y tiempo de producción en la industrialización de productos.
- El aumento del factor calidad de la producción y optimización de los recursos.
- Que el desarrollo industrial esté basado en nuevas características técnicas, especialmente en la eficiencia, flexibilidad y fiabilidad en el trabajo.

El objetivo del desarrollo y utilización de robots industriales se hace con fines de mejorar la eficiencia en los procesos de manufactura por lotes y para relevar a los operadores humanos en puestos de trabajo peligrosos e insalubres, o simplemente para efectuar tareas imposibles de realizar mediante trabajo manual.

Un robot industrial es un brazo mecánico o manipulador programable, equipado con un mínimo de capacidades sensoriales, capaz de mover piezas o herramientas a lo largo de trayectorias previamente definidas por el operador. Su programabilidad y su sistema sensorial le confieren versatilidad para desarrollar tareas variadas y adaptabilidad para reaccionar a algunos cambios en su medio ambiente.

En todas las fases de desarrollo de un robot o de un puesto de trabajo robotizado, es necesario contar con modelos apropiados para apoyar el diseño y el análisis de

tales máquinas y de cada uno de sus sistemas. Los modelos mas utilizados para diseño, simulación y control son los que describen el movimiento del manipulador, destacando los modelos cinemático, variacional y dinámico. En este trabajo se presentaran algunos de los métodos mas utilizados para derivar las ecuaciones matemáticas que describen el movimiento del manipulador desde los puntos de vista geométrico, cinemático y dinámico.

En nuestro medio carecemos este tipo de tecnologías, encontrándose grandes niveles de atraso de años tecnológicos, frente a los países que las desarrollan y emplean es por ello, este trabajo que se continúa desarrollando, tiene por finalidad implementar un prototipo para dar una visión general y objetiva de la Robótica, su estructura, control y su programación, ampliando de esta forma los conocimientos aplicados en esta fascinante tecnología, con la esperanza de implantar y desarrollar en nuestro medio ésta y otras tecnologías de punta, que permitan generar el interés en los tecnólogos, ingenieros y empresarios peruanos, fomentando la modernización y la eficiencia en la industria, basada en la automatización y desarrollo científico, contribuyendo a subsanar y cubrir de alguna manera estos vacíos de técnicas modernas carentes en nuestro país.

1.2 Definición de robot

Un robot es considerado en general, como una cadena cinemática abierta, de tracción regulada propia, que realiza un trabajo, una acción predeterminada o tareas, basado en el conocimiento de su entorno, herramientas y procesos interactivos a través de su diseño y sensores montados en él, con o sin intervención directa del hombre en el lazo de control.

Un robot para ser considerado como tal, debe cumplir ciertas características mínimas tales como:

- Aproximarse a las características humanas de trabajo.
- Tener movimiento (auto)regulado de sus efectores.
- Estar dotado de mecanismos sensoriales.
- Tener un control en lazo remoto o directo.
- Contar con un volumen de trabajo determinado.
- Ejecutar repetitivamente acciones determinadas.
- Presentar flexibilidad al intercambio de acciones.
- Ser programable y de fácil adaptación a posibles variaciones de su entorno.

Existen varias definiciones de un robot, formuladas por diversas asociaciones científicas. Cada autor, científico o tecnólogo lo define desde su punto de vista y de acuerdo a las propiedades del robot. *Webster* define a un robot como:

“Aparato o dispositivo automático que realiza funciones ordinariamente atribuidas a los seres humanos operando con una aparente inteligencia humana”.

Sin embargo, la definición que da *Robot Industries Association (RIA)* es la aceptada por la mayoría de los expertos y especialistas en el tema y lo define de la siguiente manera:

Manipulador automático, reprogramable, multifuncional o polivalente, con servosistemas de posición, capaz de posicionar y orientar materiales, piezas, útiles o dispositivos especiales a lo largo de movimientos variables y programables para la ejecución de tareas variadas.

1.3 Clasificación de los robots

En la actualidad existen diferentes clases de robots, desde los más simples hasta los más complejos, dotados de una inteligencia artificial en un conjunto específico de acciones requeridas para realizar tareas complejas y afrontar situaciones difíciles. Cuentan con una amplia versatilidad, modularidad, intercambiabilidad, existen robots móviles con uno o más brazos y en el futuro los robots podrán ver, oír, palpar, aplicar una fuerza media con precisión a un objeto y desplazarse por sus propios medios, serán capaces de responder a órdenes dadas con voz humana, contando así con muchos de los atributos de seres humanos. La clasificación de los robots se hace de acuerdo a diversos criterios, ya sea por su aplicación, por su estructura y configuración mecánica, por el tipo de control, por sus funciones de desarrollo, etc.

Hay un número de dispositivos que utilizan cierta tecnología de robots y frecuentemente en forma equivocada se les denominan robots. De hecho, Engelberg se ha referido a ellos como "relaciones cercanas" a los robots. Cuatro de tales mecanismos son:

Prótesis.

Se refieren a "brazos robóticos" o "piernas robóticas"; los cuales no pueden hacer uso de sus actuadores hidráulicos o neumáticos, no usan servo control y no tienen un sistema articulado mecánico. Ellos no tienen sus propios "cerebros" y no son realmente programables. El impulso para producir alguna acción en un dispositivo (llamada "señal de comando"); se origina en el cerebro del ser humano, esta señal es transmitida vía los nervios al accesorio apropiado, donde electrodos especiales captan los impulsos de los nervios. Estos son procesados electrónicamente por el

computador de propósitos especiales (en la tarjeta de la prótesis) los cuales convierten los controles del movimiento del miembro sustituto (mano o pierna). Hay un número de serios esfuerzos de investigación para producir tales miembros. Existen muchos problemas difíciles que restan para resolver la dificultad de extraer formalmente (de todas las actividades eléctricas producidas por los músculos vecinos) y de procesar las señales de impulso de los nervios de bajo nivel con el fin de controlar el miembro perdido.

Mecanismos dermato - esqueletos.

Son una colección (sistema articulado) de eslabonamientos mecánicos que son hechos para circundar miembros humanos o el esqueleto entero del ser humano. Estos tienen la capacidad de amplificar la potencia humana. Sin embargo, no pueden actuar independientemente y por lo tanto no son robots. De hecho, cuando un dispositivo dermato - esqueleto es usado, el operador debe ejercer extrema precaución, debido al incremento de fuerzas y/o velocidades que son posibles.

Dispositivos telequéricos.

Estos dispositivos permiten la manipulación o movimiento de materiales y/o herramientas que son localizados a una distancia considerable del operador. Estos no son robots porque requieren de un ser humano para cerrar el lazo completo y tomar las decisiones apropiadas acerca de la posición y velocidad. Tales dispositivos son especialmente usados en la manipulación de sustancias peligrosas, desechos radiactivos, para realizar exploraciones bajo el mar, etc.

Mecanismos locomotores.

Son dispositivos que imitan a los seres o animales que tiene la capacidad de caminar en dos o cuatro piernas. Aunque los múltiples accesorios pueden ser altamente sofisticados, se requiere una colección de eslabonamientos que son hidráulica o eléctricamente accionados bajo control de lazo cerrado. Un operador humano ejecuta el proceso de locomoción y toma las decisiones concernientes a la dirección deseada del dispositivo y a la coordinación del movimiento de las piernas para lograr algún objetivo.

Existen diversas maneras de clasificar a los robots. *Engelberg* sugiere clasificarlos según sus aplicaciones, su configuración mecánica y según su método de control.

A) Según las aplicaciones del robot

De acuerdo a sus aplicaciones, los robots se pueden clasificar en los siguientes campos:

Robots militares, son aquellos usados como sistemas bélicos, tanto en estaciones estáticas como en unidades móviles, en misiles, etc.

Robots espaciales, como son las naves autómatas de investigación, las ondas interplanetarias, los brazos teledirigidos, etc.

Robots submarinos, empleados para la investigación del fondo, la flora y fauna marina, para el rescate y para el mantenimiento de bases.

Robots subterráneos, empleados en las perforaciones y excavaciones de minas y túneles.

Robots experimentales, generalmente son de uso científico e investigación que se vienen desarrollando en los centros de investigación.

Robots educativos, empleados para la enseñanza de su programación y su funcionamiento.

Robots de uso domestico, empleados para realizar tareas sencillas y repetitivas en el hogar.

Robots de simulación, empleados en la medicina, en pruebas de choque de automóviles, en la aviación.

Robots industriales, para uso como manipuladores estáticos o dinámicos el la industria, mecánica, electrónica, plástica, etc.

B) Según la configuración mecánica del robot.

Aunque los mecanismos que constituyen un robot manipulador pueden variar considerablemente, todos los robots deben ser capaces de mover una carga hacia algún punto del espacio. Los ejes mayores del dispositivo normalmente consisten de dos o tres articulaciones o grados de libertad y son mecánicamente los más robustos (normalmente localizados cerca de la base), para este propósito.

Es importante entender que la clasificación cuenta sólo los ejes mayores del manipulador. Sin embargo, un robot no es limitado sólo a tres grados de libertad. Normalmente una muñeca es fijada en el extremo final del brazo. Este accesorio es, así mismo, capaz de lograr varios movimientos adicionales.

La mayoría de los robots industriales recaen en una de las siguientes cuatro categorías con respecto al sistema de coordenadas: Cartesianas, cilíndricas, esféricas y angulares las cuales son empleadas en el diseño de estos ejes. La clasificación según la configuración mecánica se puede ver en la Fig. (1.1).

i) Robots de coordenadas cartesianas

Es el tipo más simple de las configuraciones, los enlaces del manipulador son restringidos para moverse en una manera lineal. Los ejes de un dispositivo robótico que proceden en este tipo son como "prismáticos". Existen dos tipos de esta clase de robots:

Cantilevered cartesian (Cartesianos tipo viga voladiza). Aquí el brazo es conectado a un tronco, el cual está fijo a la base. Los miembros del robot manipulador son diseñados para moverse en direcciones paralelas a los ejes x, y, z. Dispositivos como éstos tienden a tener una extensión limitada desde su estructura soporte, son menos rígidos pero tienen un espacio de trabajo menos restringido que otros robots. Estos robots tienen buena repetitividad y buena precisión y son más fáciles de programar porque que el sistema de coordenadas es "más natural". Cierta tipo de movimientos puede ser más difícil de realizar con esta configuración, debido a la cantidad significativa de computación requerida, por ejemplo, una línea recta en una dirección no paralela a ningún eje.

Gantry-style cartesian (Cartesianos estilo caballete). Normalmente son usados para mover cargas extremadamente pesadas. Tales robots son frecuentemente montados en un techo interior. Ellos son generalmente más rígidos, pero pueden proporcionar menos acceso al espacio de trabajo. Para este caso, una construcción reticulada se usa para soportar al robot, de modo tal que se puede montar en un techo interior.

ii) Robots de coordenadas cilíndricas

Cuando un brazo horizontal (o bombeante) es montado en una columna

vertical y esta columna es luego montada en una base rotatoria. Con esta configuración el brazo tiene la capacidad de moverse hacia afuera y hacia adentro (en la dirección radial r). El soporte se mueve hacia arriba y hacia abajo en la columna (en la dirección z) y, el brazo y soporte ensamblados pueden rotar como una unidad en la base. Usualmente, una rotación de 360° no es permitida, debido a las restricciones impuestas por las conexiones o líneas hidráulicas, eléctricas o neumáticas. También, hay una mínima y una máxima extensión, debido a los requerimientos mecánicos. Por consiguiente, el volumen total o ambiente de trabajo es una porción de un cilindro.

iii) Robots de coordenadas esféricas

Cuando un manipulador tiene un parecido a la torreta de un tanque, es clasificado como un dispositivo de coordenadas esféricas. El brazo puede moverse entrando y saliendo y es caracterizado como un telescopio bombeante. Puede pivotear en un plano vertical y rotar en un plano horizontal alrededor de la base. Dependiendo de las limitaciones de conexiones mecánicas y/o del actuador, el ambiente de trabajo de tales robots es una porción de esfera.

iv) Robots de coordenadas angulares

Estos robots también reciben el nombre de robots de brazo articulado. Existen actualmente diferentes tipos de robots de brazo articulado. Entre éstos tenemos:

Articulados esféricos puros. Este es el más común de configuración articulada. Todos los enlaces son pivoteados, en consecuencia, pueden moverse de una manera rotatoria o de "revolución". La mayor ventaja de este diseño es que es posible llegar

cerca de la base del robot y sobre cualquier obstáculo que esté en su espacio de trabajo. La porción superior del brazo es conectada a la porción más baja (o antebrazo). El punto de pivote es frecuentemente referido a una articulación como "codo" que permite rotar el antebrazo. La parte superior del brazo es conectada a la base (a veces al tronco). El movimiento en un plano perpendicular a la base es posible en su articulación hombro. La base o tronco es también libre de rotar. De esta manera permite el ensamblamiento total moverse en un plano paralelo en la base. El ámbito de trabajo de un robot que presenta esta configuración es aproximadamente esférico.

Articulados paralelográmicos. Aquí el miembro simple rígido y superior del brazo es reemplazado por un arreglo múltiple de enlaces cerrados en la forma de un paralelogramo. La mayor ventaja de esta configuración es que permite que los actuadores de articulación sean colocados cerca o sobre la base del mismo robot. Esto significa que ellos no se mueven dentro o sobre el antebrazo o de la parte superior del mismo brazo, así que la inercia y peso del brazo son considerablemente reducidos. El resultado es una capacidad de carga grande que es posible en un dispositivo esférico articulado para el mismo tamaño de actuadores. Otra ventaja de esta configuración es que se produce un manipulador mecánicamente más rígido que los otros. La desventaja del arreglo paralelográmico es que el robot tiene un espacio de trabajo limitado comparado a un robot esférico articulado.

Articulados cilíndricos. En esta configuración, el miembro simple del eje r en un dispositivo cilíndrico puro es reemplazado por una cadena cinemática abierta de múltiple enlace. Este tipo de robots tienden a ser precisos y rápidos pero tendrán generalmente un alcance vertical limitado. Frecuentemente el movimiento en el eje z

es controlado usando un simple cilindro de aire o usando motores de paso, mientras que los otros ejes usan actuación eléctrica más elaborada (servomotores y regulación).

Una subclase de los manipuladores cilíndricos articulados es el tipo de robot **Selective Compliance Assembly Robot Arm (o SCARA)**. Típicamente, estos dispositivos son relativamente baratos y son usados en aplicaciones que requieren movimientos suaves y rápidos. Es extremadamente útil en operaciones de ensamblaje que requieren inserciones de objetos en agujeros (esto es pernos o clavos). Por construcción, el SCARA es extremadamente rígido en la dirección vertical, pero presenta elasticidad lateral (esto es docilidad), Además que facilita el proceso de inserción.

C) Según el método de control del robot.

Desde el punto de vista de la técnica usada para controlar los ejes del robot, se consideran dos casos: robots sin servocontrol y robots servocontrolados.

i) Robots sin servocontrol

También se les llama robots de secuencia limitada. Desde el punto de vista de control, los robots no servo controlados de secuencia limitada es el tipo más simple. Otros nombres frecuentemente usados para describir a estos manipuladores son robots de punto final, robot de coger y colocar (o bang - bang robot). Indiferente de la configuración mecánica o uso, la mayor característica de tales dispositivos es que sus ejes permanecen en movimiento hasta que cada uno de los límites de carrera (o parada final) se haya alcanzado. Así que solamente son asumidos dos posiciones para

los ejes individuales. También se llaman robots de posicionamiento. La naturaleza no servo implica que una vez que el manipulador ha empezado a moverse, continuará hasta que el fin de carrera apropiado sea alcanzado. No habrá monitores (vía sensores) del movimiento en cualquier punto intermedio, debido que en esta clase de robots el control se realiza de una manera de lazo abierto.

La "programación" de un robot de una secuencia limitada se concluye al colocar una secuencia deseada de movimientos y ajustar las paradas para cada eje de articulación en forma adecuada. El "cerebro" del robot manipulador consiste de un controlador/secuenciador. La porción del secuenciador es generalmente un controlador del dispositivo rotatorio (similar al temporizador del motor) con un número de contactos eléctricos. Cada contacto habilitado causa la potencia para conmutar a los actuadores de los ejes. Los ejes energizados continuarán su movimiento hasta que su programación se haya realizado. Esta información es entonces usada para causar que el índice vaya hacia el próximo paso en el programa. Es importante entender que solamente el tiempo de información es retroalimentado para el secuenciador.

ii) Robots servo-controlados

Los robots servo controlados son normalmente subdivididos en dispositivos de trayectoria continua y dispositivos punto a punto. En cualquier caso, la información acerca de la posición y de la velocidad (y tal vez otras cantidades físicas) es continuamente monitoreadas y retroalimentadas al sistema de control asociada con cada una de las articulaciones del robot. Consecuentemente, cada lazo del eje es "cerrado". El uso del control en lazo cerrado permite a los miembros del manipulador

ser comandados para moverse y detenerse en cualquier parte sin los límites de carrera para los ejes individuales. En suma, es posible el control de la velocidad, aceleración, desaceleración y sacudida (esto es, la derivada de la aceleración respecto del tiempo) para los varios ejes entre los puntos extremos. Los robots servocontrolados también tienen las siguientes características y/o atributos.

- Necesita una mayor capacidad de memoria que en los robots no servo controlados. Esto implica que son capaces de almacenar más posiciones (o puntos en el espacio). Significa también que más que un programa puede ser creado y almacenado, permitiendo de este forma que el robot pueda ser usado en una variedad de aplicaciones con un mínimo de tiempo requerido para los cambios.
- El extremo del manipulador se puede mover en cualquier de las tres diferentes clases de movimientos:
 1. Punto a punto, donde los puntos extremos del movimiento son importantes mas no la trayectoria de conexión.
 2. En línea recta, donde es importante para obtener una localización especificada en le manipulador, frecuentemente referido al punto de la herramienta, mover desde el punto inicial al punto final en forma de una línea recta en un espacio tridimensional.
 3. Trayectoria continua, donde los puntos a lo largo de la trayectoria son conectados de tal manera que la posición instantánea y su velocidad sean continuas. Si los límites impuestos por componentes mecánicos del arreglo posicional pueden ser variado ajustando las ganancias de los amplificadores apropiados en los servo lazos.
- Los actuadores de posición son usualmente dispositivos de válvula/pistón

hidráulicas, servomotores o también pueden ser servos neumáticos.

- La programación del robot generalmente se refiere al modo de enseñanza. El manipulador es movido manualmente en una secuencia deseada de puntos y las coordenadas de enseñanza de estos puntos son almacenados en la memoria del controlador del robot.
- Es posible programar cada eje del robot para mover al menos cualquier punto a lo largo de su rango total. Consecuentemente, esto produce para el usuario mayor flexibilidad en el tipo de movimientos. Además, el “movimiento coordinado” puede ser realizado mediante el movimiento simultaneo de dos o más articulaciones; así, que el extremo final del manipulador es capaz de trazar una trayectoria extremadamente compleja.
- Es posible permitir operaciones de bifurcación del manipulador basado en los datos obtenidos desde los sensores. Esta posibilidad surge del uso de microprocesadores sofisticados en el controlador del robot.
- Debido que los robots servo controlados generalmente tiene un control, un computador y estructuras mecánicas mas complejas que los dispositivos no servo controlados, son más costosos y a veces menos confiables.

Se consideran dos clase de robots servo controlados que son:

Servo - controlados punto a punto. Los robots punto a punto son ampliamente usados para mover partes u objetos de una localización a otra y también para manipular varios tipos de herramientas. Aunque éstos pueden realizar todas las tareas de los robots del tipo coge y coloca, son más versátiles que éstas por su habilidad para ser múltiplemente programados, y también, debido a la capacidad de su almacenamiento de programas.

Para esta clase de control de lazo cerrado del robot considerado aquí, solamente se indican los puntos inicial y final. La trayectoria usada para conectar los dos puntos no es importante y además no es programada por el usuario (el computador calcula la trayectoria actual del manipulador). Los robots más sofisticados punto a punto permiten movimientos en línea recta o tramos lineales. Otros también permiten que la velocidad de las articulaciones individuales sean funciones continuas en el tiempo, y también puede ser elegidas por el usuario. Esto es la velocidad con la cual el dispositivo realiza una tarea deseada, la cual es seleccionable por el usuario. Para una tarea fija, los puntos iniciales enseñados pueden ser almacenados en la memoria permanente ROM. Alternativamente, una combinación de memorias temporales RAM y ROM puede ser usadas para enseñar nuevos puntos almacenando los puntos anteriores.

En general, estos robots tienen un rango de trabajo y capacidad de carga bastante alta. Frecuentemente se usan actuadores hidráulicos. Actualmente se tiende a usar sistemas actuados por servo motores.

Servo - controlados de trayectoria continua. Muchas aplicaciones no requieren que el manipulador tenga una gran capacidad de acarrear cargas grandes. En particular, hay una clase de aplicaciones donde es más importante seguir una trayectoria compleja a través del espacio, y posiblemente, mantener el movimiento del extremo del brazo a una alta velocidad. Ejemplo de estas aplicaciones incluyen pintado a spray, pulido, molidura, esmerilado, soldadura por arco, etc. En todas estas instancias, la herramienta transportada por el manipulador puede ser bastante liviana; pero el movimiento requerido para realizar la tarea puede ser bastante complejo. Un robot de trayectoria continua es usualmente usado para estos casos.

Aunque los puntos deben todavía ser aprendidos previamente al ejecutar un programa, el método de enseñanza es usualmente diferente del usado para un robot servo controlado punto a punto. A diferencia del procedimiento descrito, los puntos no son registrados manualmente en el controlador del robot. Lo que pasa es que en el modo de aprendizaje una rutina de muestras o prueba automática es activada, la cual puede registrar puntos (y/o velocidad de información) a una razón de 60 a 80 veces en un segundo. Un operador simplemente mueve la herramienta sobre la trayectoria deseada con el ejecutor de muestreo. La razón de muestreo es lo suficientemente alta para registrar los puntos con el fin de obtener un movimiento bastante suave.

Es importante entender en general que los robots de trayectoria continua pueden ser usados solamente para un número de tareas. De otro lado, los robots punto a punto a veces tienen la habilidad de realizar un movimiento de trayectoria continua, aunque en el método de aprendizaje, debido al gran número de puntos no es muy conveniente, debido que cada punto debe ser registrado manualmente.

1.4 Especificaciones técnicas de los robots

En un robot industrial se tienen las siguientes especificaciones:

Volumen de trabajo: es el término que se refiere al espacio dentro del cual el robot puede manipular el extremo de su muñeca. El convenio de utilizar el extremo de la muñeca para definir el volumen de trabajo del robot se adopta para evitar la complicación de diferentes tamaños de efectores finales, que podrían unirse a la muñeca del robot. El efector final es una adición al robot básico y no debe contarse como parte del espacio de trabajo del robot. El volumen de trabajo viene determinado por las características físicas del robot tales como:

- Configuración física del robot, determinado por el volumen de trabajo, producto de la forma del volumen descrito por el tipo de coordenadas de dicha configuración.
- Los tamaños de los componentes del cuerpo, del brazo y de la muñeca. El tamaño y la forma de cada volumen de trabajo está influida por las dimensiones de los componentes del brazo.
- Los límites de los movimientos de las articulaciones del robot.

Sistema de coordenadas: esta especificación define la configuración del robot.

Ciertas configuraciones son mas útiles que otras en aplicaciones específicas.

Grados de libertad: se especifica como el número de pares articulación – elemento.

Movimiento máximo: esta especificación se refiere a la ubicación de fronteras en el espacio de trabajo del robot. Es importante notar que esto es usualmente medido en el reborde donde se monta la herramienta. En suma, las especificaciones máximas pueden implicar que algunas de las articulaciones son fijas. En consecuencia, en los valores máximos, no es posible orientar la herramienta en ciertas posiciones.

Sistemas de impulsión del robot: la capacidad del robot para desplazar su cuerpo, brazo y muñeca. Se proporciona por el sistema de impulsión utilizado para accionar el robot. El sistema impulsor determina la velocidad de los movimientos del brazo, la resistencia mecánica del robot y su rendimiento dinámico. En cierta medida, el sistema impulsor determina las clases de aplicaciones que puede realizar el robot. Los robots industriales disponibles en el mercado, están accionados por uno de los tres tipos de sistema de impulsión:

- *Impulsión hidráulica.*- Suele estar asociada con robots más grandes. La ventaja

habitual de este sistema es proporcionar al robot una mayor velocidad y resistencia mecánica.

- *Impulsión eléctrica.*- Este tipo de impulsión no suele proporcionar tanta velocidad o potencia como los sistemas hidráulicos, pero la exactitud y la repetibilidad suelen ser mejores. En consecuencia, los robots impulsados eléctricamente tienden a ser más pequeños, con menos exigencias de espacio y sus aplicaciones tienden hacia un trabajo más preciso, tal como el montaje.
- *Impulsión neumática.*- Suele reservarse para los robots más pequeños que tienen menos grados de libertad (dos a cuatro grados de libertad). Estos robots suelen estar limitados a simples operaciones de "coger y situar" con ciclos rápidos.

Velocidad de movimiento: Es la capacidad de velocidad medida en la muñeca de los robots industriales actuales y llega hasta un máximo de aproximadamente 1.1 m/s. En consecuencia, altas velocidades pueden obtenerse con robots grandes con el brazo extendido a una distancia máxima del eje vertical del robot. La velocidad determina la rapidez con la que el robot puede realizar un ciclo de trabajo determinado. Esto suele ser deseable en la producción para hacer mínima la duración del ciclo de una tarea dada. La determinación de la velocidad óptima, además de un simple intento para reducir al mínimo el tiempo del ciclo de producción, depende de otros factores tales como la exactitud con la que debe situarse el efector final, el peso del objeto que manipula y la distancia a recorrer. Suele existir una relación inversa entre la exactitud y la velocidad de los movimientos del robot.

Velocidad de respuesta y estabilidad: son características importantes del rendimiento dinámico en relación con el diseño de los sistemas de control. La velocidad de respuesta se refiere a la capacidad del robot para desplazarse a la

siguiente posición en un breve periodo de tiempo. La estabilidad se suele definir como la medida de las oscilaciones que se producen en el brazo durante el movimiento desde una posición a la siguiente.

Capacidad de transporte de carga: está determinada por el tamaño, la configuración, la construcción y el sistema de impulsión. Esta capacidad de carga debe especificarse bajo la condición de que el brazo del robot esté en su posición más débil. En el caso de una configuración polar, cilíndrica o de brazo articulado, significa que el brazo del robot esté en la extensión máxima.

Repetibilidad: está relacionada con la capacidad del robot para situar el efector final unido a la muñeca en un punto del espacio que se hubiera "enseñado" al robot con anterioridad. La repetibilidad también se refiere a la capacidad del robot para volver al punto programado cuando se le ordena que lo haga.

Resolución espacial: se define como el más pequeño incremento de movimiento en el que el robot puede dividir su volumen de trabajo. La resolución espacial depende de muchos factores: la resolución de control del sistema y las inexactitudes mecánicas del robot. La resolución de control viene determinada por el sistema de control del robot y su sistema de medida de retroalimentación..

Exactitud: se refiere a la capacidad de un robot para situar el extremo de su muñeca en un punto de destino deseado dentro del volumen de trabajo. La exactitud de un robot puede definirse en términos de resolución espacial, porque la capacidad para alcanzar un punto determinado de destino depende de cuán próximos pueda el robot definir los incrementos de control para cada uno de los movimientos de las articulaciones. En el caso más desfavorable, el punto deseado estaría a medio camino entre dos incrementos de control adyacentes. La exactitud varía dentro del volumen

de trabajo, tendiendo a ser peor cuando el brazo esté en el rango exterior de su volumen de trabajo y mejor cuando el brazo esté más próximo a su base. La exactitud se mejora si el ciclo de movimiento está restringido a un margen de trabajo limitado. Los errores mecánicos tenderán a reducirse cuando el robot esté sometido a un rango restringido de movimientos.

El control: define el método usado para controlar los ejes, ya sea el movimiento de trayectoria continua o el de punto a punto. Varias aplicaciones pueden requerir una o más de estas características.

Dispositivos de memoria: define ambos tipos de memoria interna y define la capacidad de almacenamiento.

Método de programación: define el tipo de adiestramiento del robot vía un teclado, a modo de enseñanza, marcha continua (en caso de pintura), disposición mecánica (para unidades no servo), o Diseño Asistido por Computador y Manufactura Asistido por Computador (CAD/CAM) que viene a ser una programación fuera de línea.

Capacidad de memoria: proporciona una indicación del número de pasos del programa o puntos que el controlador del robot puede manipular. Típicamente cada posición espacial hacia la cual el manipulador se mueve requiere un paso. Si se usan entradas/salidas, deben ser incluidas en el mismo paso o requerir otro paso adicional. También algunos robots no definen sus requerimientos de memoria en la misma manera. Por tanto es importante tener un entendimiento claro de que esta especificación realmente significa y las limitaciones que hay que considerar en la programación, así como los métodos empleados en el control del robot.

1.5 Componentes principales de un robot

Aunque la estructura mecánica, electrónica y computacional de los robots pueden variar considerablemente, se pueden destacar los componentes más comunes a un robot:

1.5.1 El Manipulador

Es una colección de eslabones mecánicos conectados por articulaciones para formar una cadena cinemática abierta. También son incluidos los elementos de transmisión de potencia (engranajes, cadenas, tornillos sin fin, etc.), elementos de reducción de fuerzas friccionales, etc. El manipulador es capaz de moverse en varias direcciones y es utilizado para hacer "el trabajo" del robot. El término "manipulador" y "robot" son frecuentemente usados recíprocamente, aunque, estrictamente hablando, esto no es correcto. Generalmente, las articulaciones de un manipulador pueden ser de revolución, y producen un movimiento puramente rotacional, o de traslación, donde producen un movimiento lineal. Cada una de las articulaciones de un robot define un eje de articulación, alrededor o a lo largo de la cual el enlace particular rota o se desliza. Cualquier par articulación - enlace o eje de articulación define un grado de libertad. Indistinto de su configuración mecánica, el manipulador definido por su estructura articulación - enlace generalmente, contiene tres principales elementos estructurales que son:

Brazo. - Conformado por los ejes mayores para el posicionamiento.

Muñeca. - Conformados por los ejes menores para la orientación.

Efactor Final .-Conformado por pinzas, herramientas, u otros objetos.

1.5.2 Actuadores

Componentes o dispositivos para accionar a las articulaciones y producir el movimiento de los miembros mecánicos. La clasificación de los actuadores se realiza basándose en su energía de activación: eléctrica, neumática e hidráulica, por lo que los actuadores pueden ser eléctricos, neumáticos e hidráulicos.

A) Actuadores eléctricos

Son elementos electromecánicos que utilizan energía eléctrica que es más manejable y que precisa instalaciones más simples.

Los motores eléctricos permiten un control previo de posición y velocidad mediante controladores electrónicos y su conexión a un autómata programable es simple. Estos actuadores son los más utilizados en el accionamiento de robots.

Los motores eléctricos se pueden clasificar en:

Motores de corriente continua DC. Son motores que reciben potencia eléctrica de corriente directa y producen una rotación mecánica de su eje. El atractivo de los motores DC está en que su velocidad es fácilmente ajustable. La mayoría de los motores DC aceleran o frenan a medida que se varía el voltaje aplicado. La desventaja principal de los motores DC es que requieren un proceso de conmutación mecánica mediante un arreglo de conmutador y escobillas. En los motores sin escobillas, la conmutación se efectúa electrónicamente. Estos motores se dividen a su vez en motores de rotor devanado y motores de imán permanente.

Motores de corriente alterna AC. Son motores que se conectan directamente a la línea de corriente alterna. Los motores AC tienen la ventaja de no necesitar un proceso de conmutación para poder operar, y tampoco requieren un suministro de

potencia rectificada de DC. Su desventaja principal es que su velocidad no es fácilmente ajustable, ya que está atada fuertemente a la frecuencia de la línea de AC. Los motores AC a su vez se clasifican en motores AC de inducción, motores AC síncronos y motores de AC universales.

Motores paso a paso. Son motores síncronos y sin escobillas y de conmutación electrónica, que a partir de una señal de mando en forma de impulsos, giran un cierto ángulo, cuyo valor es un número entero de veces un ángulo elemental. Debido a esta característica, este tipo de motores es especialmente adecuado para el control en lazo abierto, sin retroalimentación. Los motores paso a paso trabajan con grandes fluctuaciones instantáneas de velocidad, lo que ocasiona aceleraciones elevadas. En estos motores la acción de conmutación necesaria para realizar su función se logra por transistores externos. Es más, el rotor no tiene devanado de armadura; simplemente es una colección de imanes permanentes salientes. La principal ventaja es su velocidad ajustable como una función de un tren de impulsos aplicados a sus devanados. Estos motores mantienen su torque aún cuando no haya conmutación. Sus desventajas son su baja velocidad, baja potencia y baja precisión en posicionamiento. Estos motores se dividen en motores de rotor de imán permanente y en motores de reluctancia variable.

B) Actuadores neumáticos

La fuente de energía de este tipo de elementos es aire a presión, y dado que el aire es compresible, el posicionamiento dependerá de la carga de trabajo. Por ello la precisión de la trayectoria continua que no puede obtenerse con este tipo de elementos será inferior a la obtenible con los otros tipos de actuadores, a menos que

se utilicen sensores de posición. Los elementos motores neumáticos son básicamente de dos tipos:

Motores neumáticos. Los motores neumáticos son ligeros y compactos. El arranque y parada son rápidos y pueden trabajar con velocidad y par variables sin necesidad de un control complejo. En el caso de motores reversibles, puede cambiarse el sentido de giro sin producir sacudidas sobre la carga. Trabajando en vacío, el cambio de sentido puede realizarse en muy pocos grados gracias a su alta aceleración y baja inercia. Los motores de aire más adecuados para su empleo en Robótica son los rotativos en general, con elemento reductor, utilizándose también los de pistón radial y axial.

Cilindro neumático. Se diferencia del motor neumático en que el movimiento de actuación es rectilíneo. En general se trata de cilindros de doble efecto, con potencia para el movimiento en ambas direcciones, controlados por una o más válvulas de distribución, o también por servoválvulas, siendo posible efectuar un buen control de posición y velocidad.

C) Actuadores hidráulicos

Son actuadores cuya fuente de energía es hidráulica. Son análogos a los neumáticos. El fluido es aceite y presenta características distintas al aire. Así como el aire es compresible y de baja viscosidad, el aceite es incompresible a la presión de trabajo y más viscoso, haciendo adecuado el accionamiento hidráulico para movimientos que no sean excesivamente rápidos, pero que requieran una mayor precisión y repetitividad manejando cargas importantes.

Dentro de los actuadores hidráulicos se consideran los motores y los cilindros.

Motores hidráulicos. Son dispositivos simples y robustos, soportando sobrecargas sin consecuencias posteriores. La velocidad de rotación es regulable en los dos sentidos y la inversión del sentido de giro es simple. Manteniendo constantes el caudal y la presión, podemos aumentar la velocidad disminuyendo la cilindrada. En general los motores hidráulicos pueden ser de engranajes, de aletas y de pistones.

Cilindros neumáticos. Son dispositivos por lo general de doble efecto, va siempre asociado a una válvula que controla el flujo del fluido a una cara u otra del pistón, provocando la diferencia de presiones necesaria para el desplazamiento, y constituye un buen elemento accionador para un sistema de servocontrol que requiera desplazamientos lentos y potentes.

1.5.3 Sensores

Estos elementos informan al controlador del robot acerca del estado del manipulador. Esto puede ser hecho continuamente o solamente al final de cada movimiento deseado. Por ejemplo en algunos robots, los sensores proporcionan información acerca de la posición, velocidad y fuerzas instantáneas alrededor de los enlaces individuales que pueden ser retroalimentados en la unidad de control para producir el control apropiado del sistema mecánico. En el caso más simple, el controlador puede ser informado sólo cuando los enlaces individuales del manipulador han realizado o alcanzado sus fines programados o sus posiciones finales. Indistintamente de cómo es usada, la información proporcionada por los sensores puede ser bien analógica, digital o una combinación. Los sensores en los robots modernos se pueden dividir en sensores internos y sensores externos.

A) Sensores internos

Son los sensores usados por el robot para controlar su posicionamiento, movimiento, velocidad, aceleración y/o fuerza

De posición, entre éstos se tiene los codificadores ópticos, potenciómetros, resolvidores, etc.

De velocidad, entre éstos se tienen los tacómetros.

De fuerza/torque, para protección de sobrecargas,

B) Sensores externos

Este tipo de sensores sirven para proporcionar "inteligencia" al robot con el objetivo de realizar alguna tarea específica. Entre estos sensores se tiene a:

No visuales. A su vez éstos pueden ser:

- **Sensores de contacto o táctiles**, útiles para el seguimiento de trayectorias, reconocimiento de partes, o para trabajos de ensamblaje.
- **Sensores de proximidad**, para evita colisiones.
- **Sensores de alcance**, entre estos se tienen a los ultrasónicos,

Visuales. Son usados para reconocimiento de objetos, guiar al robot y son:

- De visión.
- Dispositivos de acoplamiento de carga.
- Dispositivos de infección de carga.
- Cámaras de televisión.

1.5.4 Elementos de transmisión

En algunos casos, los elementos motores no pueden actuar directamente sobre el

elemento móvil debido que la masa de los motores acoplados directamente supone una mayor inercia de carga para las articulaciones anteriores y limita la máxima aceleración posible.

El sistema transmisor puede realizar un cambio en la dirección del movimiento, incluso una transformación del mismo y también puede efectuar una reducción de velocidad o de desplazamiento. Entre los elementos que se aplican en el diseño de los robots se tiene:

- **Transmisión de movimiento circular.**- Este tipo de transmisión puede conllevar una variación de la velocidad de giro que se transmite. Entre los dispositivos de este tipo se tiene: engranajes, correas dentadas, cadena, cables, biela manivela y otros.
- **Conversión de movimiento circular en lineal.**- Este tipo de conversión puede ser útil en robots con actuadores eléctricos y que efectúan movimientos lineales o para conversión de movimiento lineal en circular. Entre estos tipos de elementos se tiene: tornillo sin fin, y engranaje cremallera.
- **Conversión de movimiento lineal en circular.**- Se utilizan para el desplazamiento de un punto de un brazo que dispone de un extremo con un apoyo fijo mediante un actuador lineal sujeto por un extremo a un apoyo fijo, y por el otro al brazo mediante dos rótulas que le permiten girar. Otros métodos de conversión de movimiento lineal en circular se basan en la utilización de mecanismos articulados con una geometría de paralelogramos deformables que realizan la transmisión de movimiento con conversión del desplazamiento lineal en curvilíneo.

1.5.5 El controlador

El controlador del robot generalmente realiza tres funciones fundamentales:

1. Iniciar y terminar el movimiento de los componentes individuales del manipulador en una secuencia deseada y puntos específicos.
2. Almacenar la información de la posición y velocidad en su memoria.
3. Permitir al robot interactuar con el mundo "exterior" vía los sensores montados en el área donde el trabajo se está realizando (estación de trabajo). Para llevar a cabo esto, los controladores realizan los cálculos computacionales necesarios para la determinación de la trayectoria correcta del manipulador, velocidad y posición. El controlador envía señales a los dispositivos que accionan a las articulaciones (vía interfaces) y utiliza la información proporcionada por los sensores del robot. Finalmente, el controlador debe permitir la comunicación entre los dispositivos periféricos y el manipulador.

Los controladores usualmente caen en una de las siguientes categorías:

- Secuenciador de simple paso.
- Sistema lógico neumático.
- Secuenciador electrónico.
- Sistema de microprocesadores.
- Sistema electrónico que permite el control del robot y comprende un sistema de microprocesadores y una interfaz de entrada y salida.

1.5.6 Unidad de conversión de energía

La unidad de conversión de energía contiene los componentes necesarios para llevar una señal del secuenciador (digital o nivel analógico bajo) y la convierte en un

nivel de potencia suficiente como para hacer mover los motores. Estos elementos pueden consistir en amplificadores de potencia electrónicos, mientras que para el caso de actuadores hidráulicos, tales elementos pueden consistir de un compresor y válvulas de control.

1.6 El ROB-DEX

El *ROB-DEX* es un robot que se basa estructuralmente en el modelo *SCARA* dentro de la clasificación de robots. Este robot se caracteriza por su total originalidad y presenta las siguientes características

- ◆ Está construido mecánicamente por elementos metálicos de fierro dulce o maquinable, en su mayoría cilíndricos.
- ◆ Consta de cuatro articulaciones o grados de libertad, siendo las tres primeras angulares y la cuarta articulación traslacional.
- ◆ Se caracteriza por presentar una gran rigidez verticalmente y una gran flexibilidad horizontalmente.
- ◆ Los actuadores se basan en motores paso a paso unipolares síncronos y de imán permanente, y son los que se encargan de accionar las articulaciones.
- ◆ El sistema de transmisión de movimiento está basado en engranajes helicoidales para las articulaciones angulares y en un tornillo sin fin de doble entrada para la articulación prismática.
- ◆ Los enlaces o eslabones se deslizan a través de rodamientos cónicos con el fin de disminuir las fuerzas de fricción opuestas al movimiento de las articulaciones.
- ◆ Los sensores de estado interno se basan en decodificadores ópticos incrementales que registran las variables de desplazamiento con sus respectivas velocidades.

- ◆ Con la estructura descrita y gracias a la potencia de sus motores, se pueden manipular y transportar bloques de hasta 2 kilos de peso.
- ◆ El efector final se basa en una pinza de dedos deslizantes en los que la apertura y cierre se produce por desplazamiento paralelo de los dedos.
- ◆ El espacio o volumen de trabajo máximo, del *ROB-DEX* es de $65^2 \times 32\pi$ (cm³).
- ◆ Debido al juego en la transmisión de potencia en los engranajes, la repetibilidad y la exactitud del *ROB-DEX* son bajas.
- ◆ El uso de motores paso a paso hace que el *ROB-DEX* tenga una baja resolución y disminuya el rango de aplicaciones.
- ◆ La arquitectura electrónica se basa en tarjetas impresos elaboradas en nuestro país.
- ◆ El controlador del *ROB-DEX* es un sistema electrónico que se basa en un microprocesador 8088 y componentes electrónicos programables.
- ◆ La energía suministrada al sistema electrónico y a los motores se hace mediante una fuente conmutada de un computador personal.
- ◆ Toda la estructura mecánica llega a pesar 110 Kg.
- ◆ La programación de bajo y alto nivel se ha elaborado en base los lenguajes C++ y ensamblador.
- ◆ Cuenta con una interfaz gráfica de usuario para interactuar con el sistema de control total del robot.
- ◆ El sistema software cuenta con la simulación gráfica del robot en tiempo real.

A la simulación gráfica se le ha denominado "*alma*" del robot y a la parte física o mecánica se le ha denominado "*cuerpo*" del robot. A la estructura física total del *ROB-DEX* se puede apreciar en la Fig. (1.2).

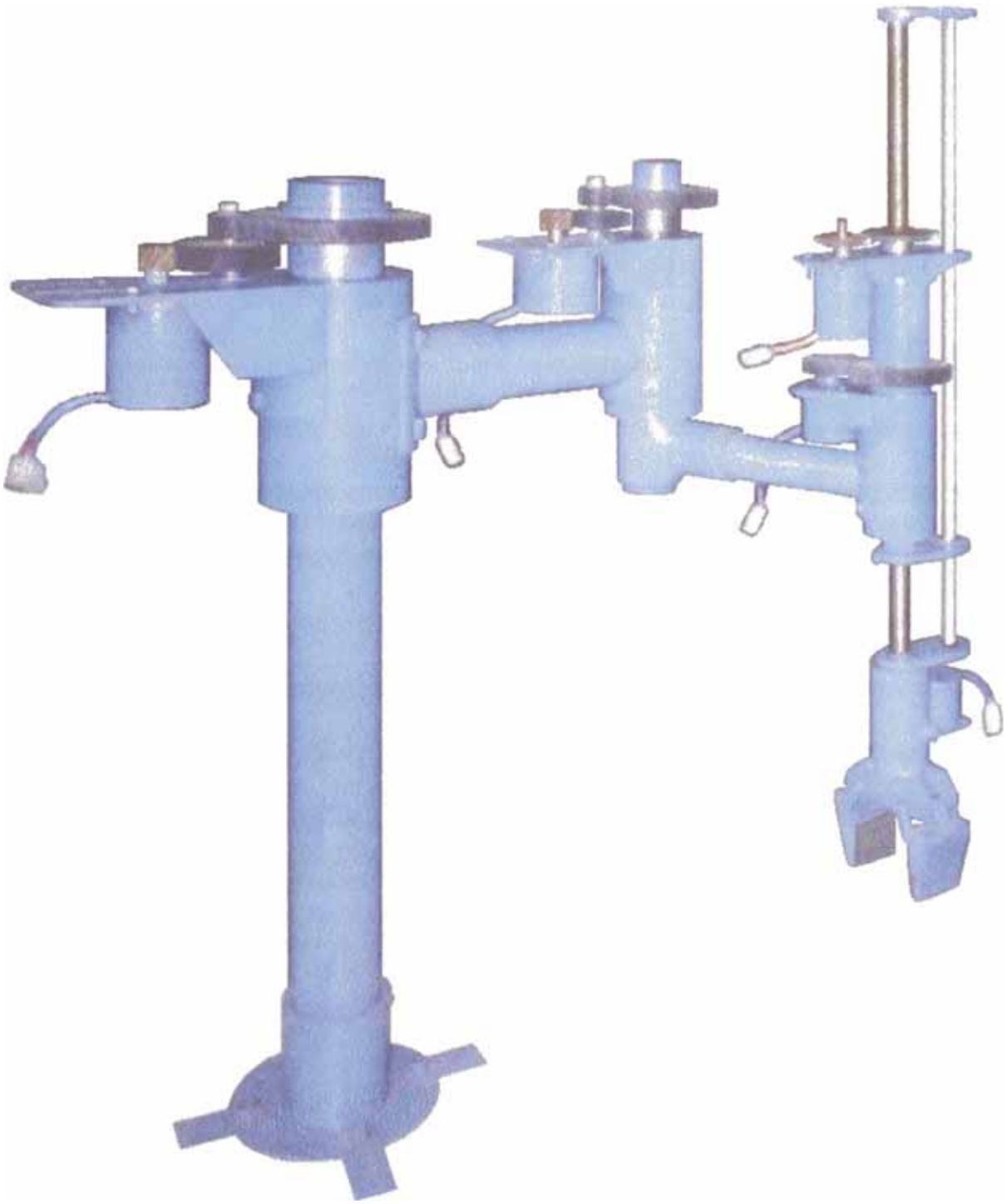


Fig. 1.2 Estructura física del *ROB-DEX*.

CAPITULO II

CINEMATICA DEL ROBOT

2.1 Introducción

La cinemática del brazo de un robot trata del estudio analítico de la geometría del movimiento espacial del robot (sin tener en cuenta las fuerzas que lo producen), con respecto a un referencial fijo denominado referencial base, a manera de una función del tiempo y de sus parámetros que definen la estructura física de cada elemento del robot, los cuales determinan la posición relativa entre elementos adyacentes o contiguos, transformando de esta manera las coordenadas de articulación a un sistema de coordenadas de referencia y viceversa. A la primera transformación se le denomina *cinemática directa* y a la segunda *cinemática inversa*, gráficamente estas transformaciones se representan en la Fig. (2.1).

A la cinemática del robot también se le llama *modelo cinemático*. Este modelo al relacionar sólo variables de posición, debería llamarse *modelo geométrico* en vez de cinemático, pues no considera la velocidad con la que se mueve el robot; sin embargo en la literatura de Robótica, se acostumbra llamarlo modelo cinemático.

En la cinemática de un robot se plantean dos cuestiones fundamentales, ambas de interés teórico y práctico, que son:

1. Para un manipulador determinado de n grados de libertad, dado el vector de variables de las articulaciones $q(t)$ y los parámetros geométricos de los elementos

o enlaces, ¿cual es la posición \mathbf{p} y la orientación \mathbf{R} del efector final del robot con respecto a un sistema de coordenadas de referencia?

2. Dada una posición \mathbf{p} y una orientación \mathbf{R} deseada del efector final del robot y conociendo los parámetros geométricos de los elementos con respecto a un sistema de coordenadas de referencia, ¿ puede el robot alcanzar la posición y orientación del efector final que se desea?. Y si puede, ¿cuales son los valores de las variables articulares $\mathbf{q}(t)$ y cuantas configuraciones diferentes del manipulador satisfacen la misma condición?

La primera pregunta se conoce como el *problema cinemático directo*, mientras la segunda es el *problema cinemático inverso* (solución del brazo).

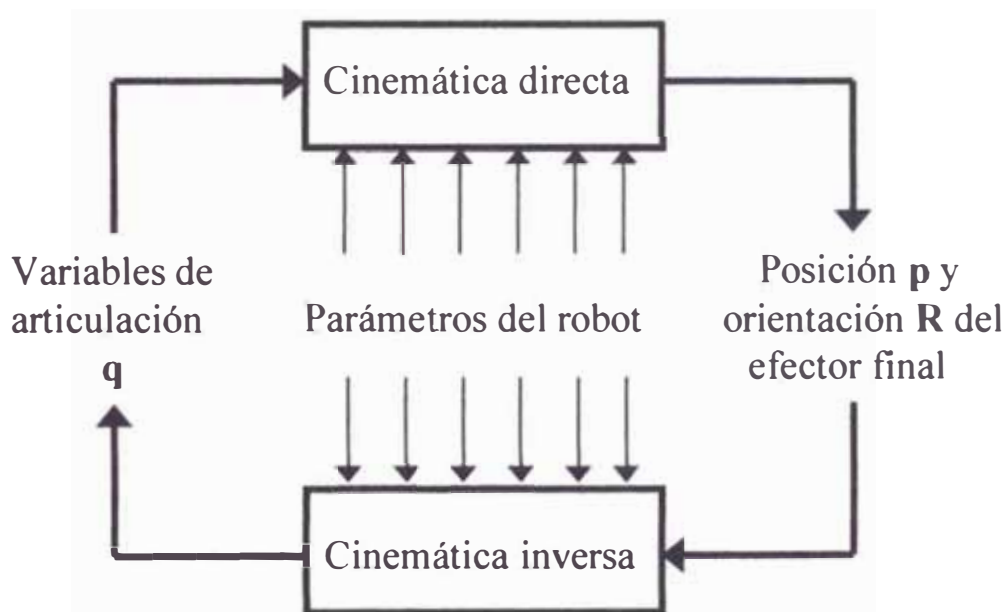


Fig. 2.1 Cinemática directa e inversa de un robot.

Como las variables independientes en un brazo de robot son las variables de articulación y una tarea se suele dar en términos, del sistema de coordenadas de la referencia, el *problema cinemático inverso* se utiliza con mayor frecuencia.

2.2 Estructura cinemática

La estructura mecánica de un robot manipulador es una cadena poliarticulada normalmente abierta, formada por una secuencia de elementos rígidos denominados también eslabones o enlaces, que están unidos uno a otro por articulaciones simples que pueden ser de rotación o de traslación. Un extremo de la cadena está fijo o anclado a una base soporte, mientras que el otro extremo está libre para moverse. En el extremo móvil está fijado un órgano terminal denominado efector final (pinza, herramienta, etc.) que se utiliza por el robot para realizar alguna tarea específica. En la Fig. (2.2) se aprecia la configuración mecánica de una cadena articulada.

Las articulaciones tienen dos superficies deslizantes, una sobre la otra, mientras permanecen en contacto, y, se diseñan para permitir al robot desplazar su efector final a una posición deseada, dentro de los límites del tamaño del robot y de los movimientos de las articulaciones.

Las articulaciones utilizadas en el diseño de los robots industriales, implican un movimiento relativo de los eslabones contiguos, movimiento que puede ser lineal o rotacional. Las articulaciones lineales implican un movimiento deslizante o de traslación de los enlaces de conexión; en robótica también se les denominan articulaciones prismáticas. Las articulaciones giratorias pueden ser de tipo rotacional, giratoria o de tipo revolución.

En un manipulador mecánico los movimientos de las articulaciones individuales se denominan grados de libertad; o sea, para un manipulador de n grados de libertad, hay n articulaciones y $n+1$ enlaces, con el enlace 0 (inmóvil) unido a la base soporte, donde se establece un sistema de coordenadas o referencial para el sistema dinámico. Cada enlace se conecta, a lo más, a otros dos, para no formar lazos cerrados.

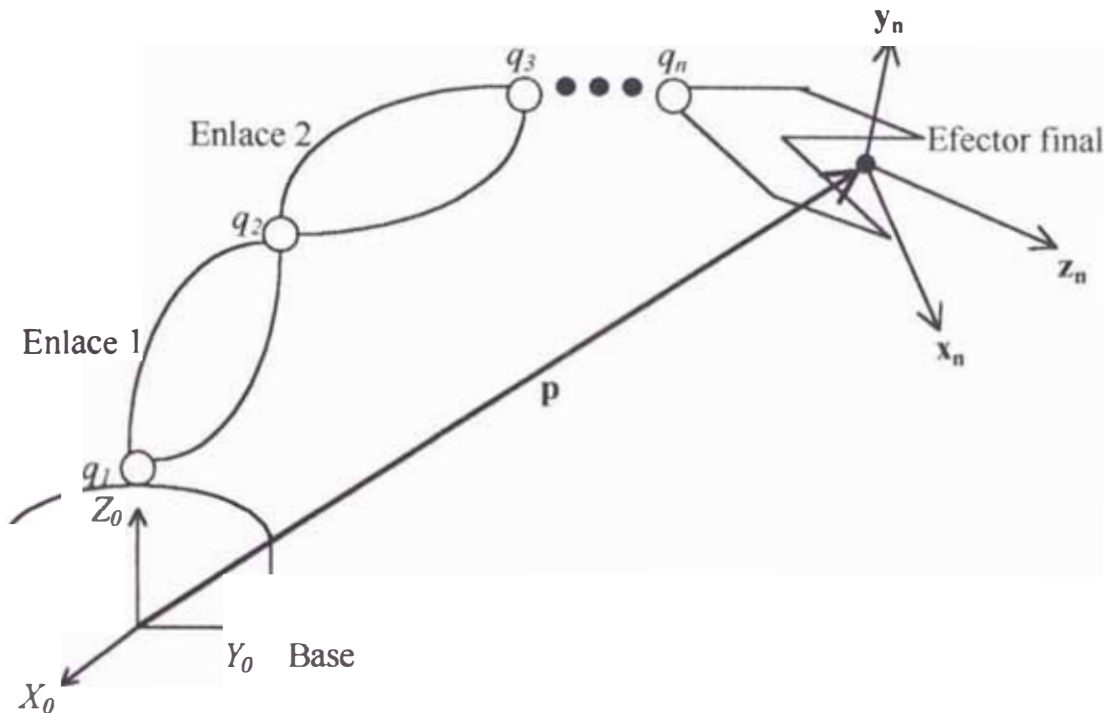


Fig. 2.2 Cadena cinemática representando la mecánica de un robot.

Para poder realizar cualquier trabajo, el robot debe contar con al menos seis grados de libertad; pues, además de posicionarse en cualquier punto de su espacio de trabajo (con al menos 3 grados de libertad), el robot debe orientar su efector final hacia cualquier dirección (con otros 3 grados de libertad). Por tal motivo, puede considerarse entonces que el robot está compuesto por un *brazo* con tres articulaciones para posicionarse y por una *mano* con tres articulaciones más, para orientarse. A la *posición* y a la *orientación* juntas se les denomina *localización*.

Si el número de articulaciones supera al número de componentes cartesianas, se dice que el robot es cinemáticamente redundante.

La cinemática de un robot, no es otra cosa que una relación entre las variables articulares y las variables cartesianas correspondientes a la posición más la orientación del efector final. La cinemática, al relacionar tan sólo variables de posición, debería llamarse *geometría del robot* en vez de cinemática, pues no

considera la velocidad con la que se mueve el manipulador; sin embargo, en la literatura especializada de Robótica, se acostumbra llamarla cinemática del robot o modelo cinemático. El modelo cinemático es muy utilizado en la simulación cinemática del robot, en la fase inicial del diseño mecánico, en el análisis de puestos de trabajo robotizados, en control (generación de trayectorias), y, en el análisis de información sensorial (visual y proximetría).

2.3 Representación de Denavit-Hartenberg

Para describir la relación traslacional y rotacional entre elementos adyacentes, *Denavit y Hartenberg (D-H)* propusieron un método matricial para establecer de forma sistemática un sistema de coordenadas (ligado al cuerpo) para cada elemento de la cadena articulada. La representación de *Denavit-Hartenberg* se expresa mediante una *matriz de transformación homogénea* T (mostrada en el Apéndice A) que representa cada uno de los referenciales de los elementos en la articulación con respecto al referencial del elemento previo. Así, mediante transformaciones secuenciales, el efector final expresado en las "coordenadas de la mano" se puede transformar y expresar en las "coordenadas de la base" que constituyen el sistema inercial del manipulador.

Considerando una cadena cinemática con n grados de libertad, cada enlace o elemento k como el mostrado en la Fig. (2.3), conecta sus extremos a otros dos elementos para establecer dos ejes de articulación z_{k-1} y z_k y mantener una configuración fija entre sus articulaciones. La distancia más corta a_k medida a lo largo de la normal común entre los ejes de articulación z_{k-1} y z_k se denomina longitud del eslabón k , y el ángulo α_k que forman estos ejes, medido en el plano perpendicular

a a_k es el ángulo de torsión del elemento k ; por lo tanto, un eslabón tiene dos dimensiones o parámetros inherentes a_k y α_k que determinan su estructura (parámetros del eslabón).

Un eje de articulación se establece para la k -ésima articulación en la conexión de dos elementos, el cual tendrá dos normales conectadas a él, una para cada uno de los elementos adyacentes a_{k-1} y a_k . La posición relativa de tales elementos conectados (elemento $k-1$ y elemento k) que viene dada por d_k , es la distancia medida a lo largo del eje de la articulación entre las normales. El ángulo de articulación θ_k entre las normales a_{k-1} y a_k se mide en un plano normal al eje de la articulación k . Los parámetros d_k y θ_k son llamados distancia relativa y ángulo entre los elementos adyacentes, respectivamente (parámetros de la articulación).

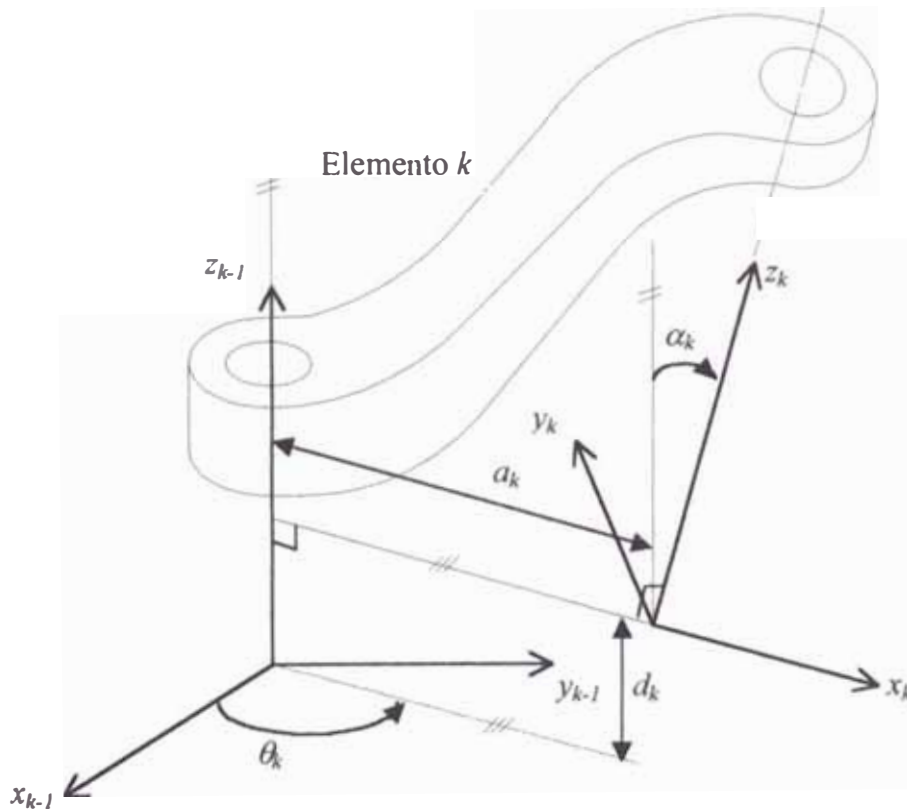


Fig. 2.3 Parámetros de un par elemento-articulación.

Con los cuatro parámetros geométricos asociados a_k , α_k , d_k y θ_k a cada par articulación - elemento de un manipulador y estableciendo un convenio de signos para cada parámetro, se puede determinar completamente la configuración cinemática de un robot manipulador mediante la representación de $D-H$ para describir completamente cualquier articulación de rotación o de traslación.

En resumen estos parámetros se pueden también especificar como sigue:

- θ_k = ángulo de la articulación del eje x_{k-1} al eje x_k respecto del eje z_{k-1} .
- d_k = distancia desde el origen del sistema de coordenadas $(k-1)$ -ésimo hasta la intersección del eje z_{k-1} con el eje x_k a lo largo del eje z_{k-1} .
- α_k = ángulo de separación del eje z_{k-1} al eje z_k respecto del eje x_k .
- a_k = distancia de separación desde la intersección del eje z_{k-1} con el eje x_k hasta el origen del sistema k -ésimo a lo largo eje x_k .

Para una articulación giratoria o rotacional, a_k , α_k y d_k son los parámetros del elemento y permanecen invariantes para el robot, mientras que θ_k es la variable de articulación que varía cuando el elemento k gira con respecto al elemento $k-1$.

Para una articulación prismática o traslacional, a_k , α_k y θ_k son los parámetros del elemento y permanecen constantes para el robot, mientras que d_k es la variable de la articulación que varía cuando el elemento k se desplaza con respecto al elemento $k-1$.

Para expresar la cinemática de cada par articulación - elemento, cuando se tiene una articulación rotacional o angular como la de la Fig. (2.4), haciendo uso de la formulación $D-H$, se determina adecuadamente un sistema de coordenadas a cada elemento del robot; por ejemplo, al eslabón $k-1$ y al eslabón k , respectivamente; entonces el sistema de coordenadas del eslabón $k-1$ es el sistema de coordenadas de

referencia y el sistema de coordenadas del eslabón k es el referencial móvil cuando se activa la articulación k . Una vez establecido el sistema de coordenadas $D-H$ para cada elemento, se desarrolla una matriz de transformación homogénea que relacione el sistema de coordenadas k -ésimo con el sistema de coordenadas $(k-1)$ -ésimo en función de sus respectivos parámetros.

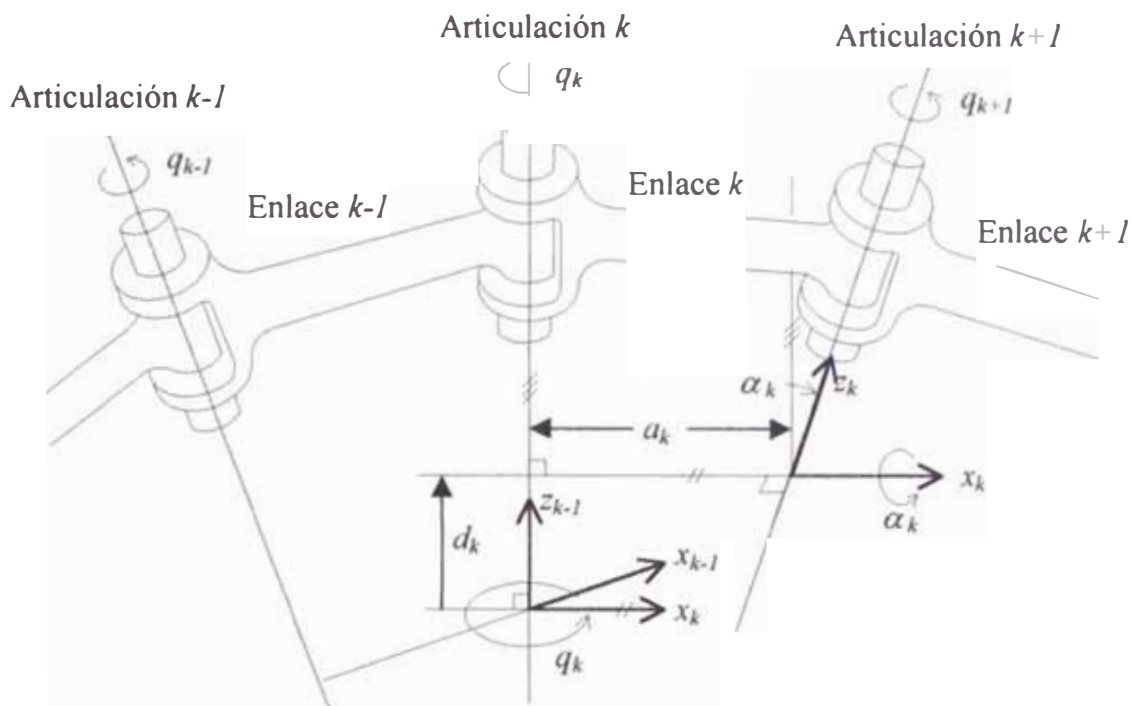


Fig. 2.4 Parámetros de una articulación rotacional.

El origen del referencial del enlace k es puesto en la intersección de la normal común entre el eje de articulación $k-1$ y el eje de la articulación k . En el caso de que se intercepten los ejes de articulación, el origen es puesto en el punto de intersección de los ejes de articulación con $a_k=0$. Si los ejes son paralelos, el origen se elige de tal manera que en la articulación, la distancia sea cero para el siguiente elemento, cuyo origen de coordenadas está definido.

Los ejes del sistema de coordenadas $0x_k y_k z_k$ o referencial L_k se seleccionan del siguiente modo:

1. El eje z_k del sistema k debería coincidir con el eje de la articulación $k+1$, al rededor del cual la rotación de q_{k+1} se está realizando.
2. El eje x_k se deberá alinear con cualquier normal común que exista (usualmente la normal entre los ejes de articulación k y $k+1$) y es direccionado desde la articulación k a la articulación $k+1$. En el caso de que los ejes de las articulaciones se intercepten, el eje x_k se elige para ser paralelo o antiparalelo al producto vectorial de $z_{k-1} \times z_k$.
3. El eje y_k debe satisfacer $x_k \times y_k = z_k$.

La coordenada de articulación rotacional q_k es ahora definido como el ángulo entre los ejes x_{k-1} y x_k es cero cuando estos ejes son paralelos y tiene la misma dirección.

Las matrices de transformación en el sistema de coordenadas del elemento o enlace y los vectores distantes entre sus orígenes se determinan mediante las matrices 4×4 de transformación homogénea. La transformación del sistema $k-1$ en el sistema k puede ser descrito por una secuencia de rotaciones y traslaciones: una rotación con respecto al eje z_{k-1} con un ángulo q_k (para una articulación de traslación con un ángulo θ_k), una traslación a lo largo de z_{k-1} de magnitud d_k (q_k para una articulación prismática), una traslación a lo largo del eje $x_{k-1}=x_k$ rotado de longitud a_k y una rotación con respecto del eje x_k para el ángulo de torsión α_k .

La secuencia de rotaciones y traslaciones para una articulación de rotación se representa como un producto de las matrices de transformación homogénea, como sigue:

$${}_{k-1}\mathbf{T}^k = Rot(z, q_k) Tras(0, 0, d_k) Tras(a_k, 0, 0) Rot(x, \alpha_k) \quad (2.1)$$

$${}^{k-1}\mathbf{T}^k = \begin{bmatrix} \cos q_k & -\text{sen } q_k & 0 & 0 \\ \text{sen } q_k & \cos q_k & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_k \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha_k & -\text{sen } \alpha_k & 0 \\ 0 & \text{sen } \alpha_k & \cos \alpha_k & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.2)$$

Multiplicando las matrices anteriores se obtiene la matriz de transformación homogénea 4×4 entre dos coordenadas sucesivas k y $k-1$ para una articulación de revolución como:

$${}^{k-1}\mathbf{T}^k \triangleq \begin{bmatrix} \cos q_k & -\cos \alpha_k \text{sen } q_k & \text{sen } \alpha_k \text{sen } q_k & a_k \cos q_k \\ \text{sen } q_k & \cos \alpha_k \cos q_k & -\text{sen } \alpha_k \cos q_k & a_k \text{sen } q_k \\ 0 & \text{sen } \alpha_k & \cos \alpha_k & d_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

Utilizando la ecuación (A.22) del Apéndice A se puede encontrar la inversa de esta transformación como:

$${}^k\mathbf{T}_{k-1} \triangleq \begin{bmatrix} \cos q_k & \text{sen } q_k & 0 & -a_k \\ -\cos \alpha_k \text{sen } q_k & \cos \alpha_k \cos q_k & \text{sen } \alpha_k & -d_k \text{sen } \alpha_k \\ \text{sen } \alpha_k \text{sen } q_k & -\text{sen } \alpha_k \cos q_k & \cos \alpha_k & -d_k \cos \alpha_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.4)$$

En el caso de una articulación prismática como la mostrada en la Fig. (2.5), el sistema de coordenadas del enlace es elegido de tal manera que la distancia d_k viene a ser la variable de articulación q_k ; el parámetro a_k no se ha tomado en cuenta y es puesto a cero, mientras que el ángulo que tubo una rotación de ángulo q_k para una articulación de revolución viene a ser un ángulo constante denotado por θ_k . El origen del sistema de coordenadas correspondiente a la articulación de deslizamiento k es elegido para coincidir con el siguiente origen del enlace definido $k+1$. El eje z_k es alineado con el eje de la articulación $k+1$. El eje x_k es elegido para ser paralelo o

antiparalelo al vector $z_{k-1} \times z_k$. El cero de la coordenada de la articulación $q_k = d_k$ es definido cuando el origen del sistema de coordenadas local $0_{k-1}x_{k-1}y_{k-1}z_{k-1}$ (referencial L_{k-1}) y $0_kx_ky_kz_k$ (referencial L_k) coinciden. El ángulo θ_k entre los ejes x_{k-1} y x_k es fijo y representa un parámetro cinemático con el ángulo de torsión α_k .

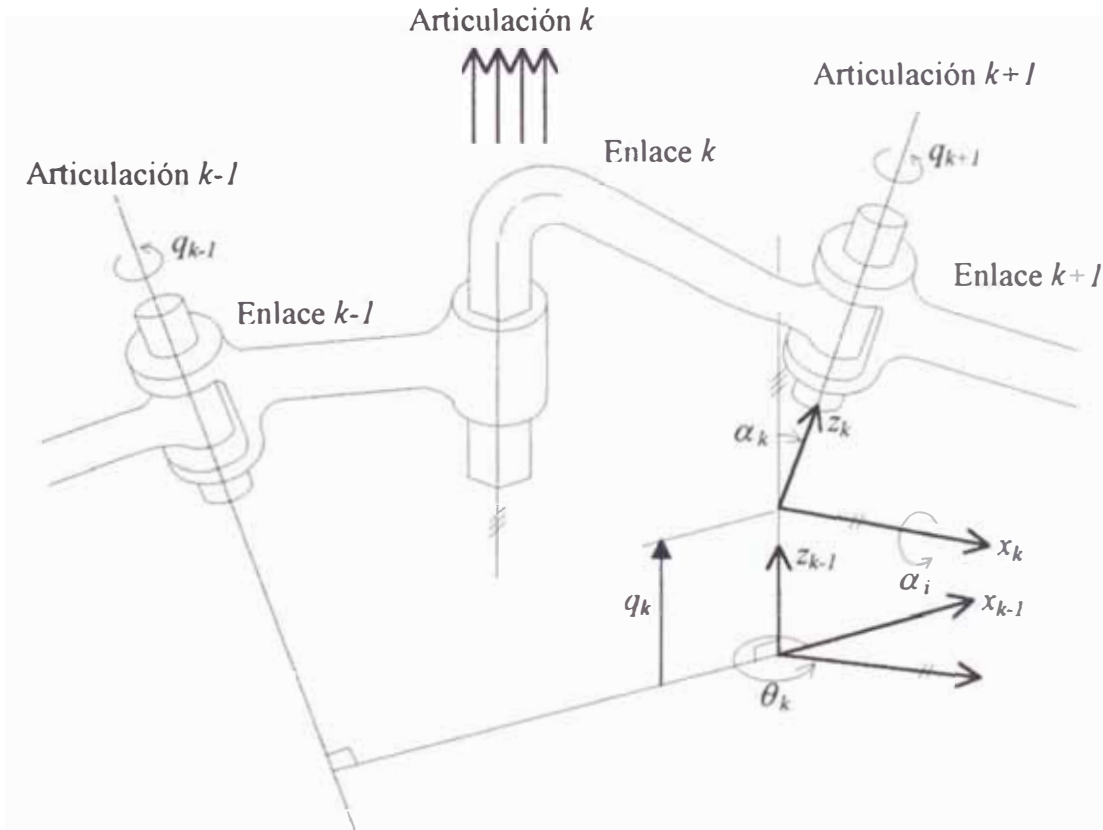


Fig. 2.5 Parámetros de una articulación traslacional.

La secuencia de rotaciones y traslaciones para una articulación de traslación se representa como un producto de las matrices de transformación homogénea, como sigue:

$${}_{k-1}\mathbf{T}^k = Rot(z, \theta_k) Tras(0, 0, q_k) Rot(x, \alpha_k) \quad (2.5)$$

$${}_{k-1}\mathbf{T}^k = \begin{bmatrix} \cos\theta_k & -\text{sen}\theta_k & 0 & 0 \\ \text{sen}\theta_k & \cos\theta_k & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_k & -\text{sen}\alpha_k & 0 \\ 0 & \text{sen}\alpha_k & \cos\alpha_k & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

Multiplicando las matrices anteriores se obtiene la matriz de transformación homogénea 4×4 entre dos coordenadas sucesivas k y $k-1$ para una articulación de traslación como:

$${}_{k-1}\mathbf{T}^k \triangleq \begin{bmatrix} \cos\theta_k & -\cos\alpha_k \sin\theta_k & \sin\alpha_k \sin\theta_k & 0 \\ \sin\theta_k & \cos\alpha_k \cos\theta_k & -\sin\alpha_k \cos\theta_k & 0 \\ 0 & \sin\alpha_k & \cos\alpha_k & q_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.7)$$

Utilizando la ecuación (A.22) del Apéndice A se puede encontrar la inversa de esta transformación como:

$${}^k\mathbf{T}_{k-1} \triangleq \begin{bmatrix} \cos\theta_k & \sin\theta_k & 0 & 0 \\ -\cos\alpha_k \sin\theta_k & \cos\alpha_k \cos\theta_k & \sin\alpha_k & -q_k \sin\alpha_k \\ \sin\alpha_k \sin\theta_k & -\sin\alpha_k \cos\theta_k & \cos\alpha_k & -q_k \cos\alpha_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.8)$$

Una vez que el sistema de coordenadas son asignadas a los enlaces de un manipulador dado y los parámetros cinemáticos a_k , α_k , d_k y θ_k , $k = 1, 2, 3, \dots, n$ son determinados, las matrices de transformación homogénea (2.3) o (2.7) dependen solamente de las coordenadas articulares. Por tanto, la forma analítica de los elementos de la matriz ${}_{k-1}\mathbf{T}^k$ es usualmente sencilla. Habiendo obtenido las matrices de transformación homogénea entre los referenciales de coordenadas sucesivas, se puede obtener fácilmente la transformación homogénea entre los referenciales asignados al último enlace y referencial de coordenadas de referencia.

2.3.1 Algoritmo para la representación D-H

Para un robot con n grados de libertad, el etiquetado de cada referencial comienza desde la base soporte hasta el efector final del brazo. La relación entre elementos

adyacentes se representa mediante una matriz de transformación homogénea ${}_{k-1}\mathbf{T}^k$.

Cada sistema de coordenadas se determina y establece sobre la base de tres reglas:

1. El eje z_{k-1} yace a lo largo del eje de la articulación.
2. El eje x_k es normal al eje z_{k-1} y apunta hacia afuera de él.
3. El eje y_k completa el sistema de coordenadas dextrógiro según se requiera.

Con las tres reglas básicas anteriores, se realiza un algoritmo para establecer un sistema de coordenadas ortonormal *consistente* en cada elemento del robot de la siguiente manera:

1. *Establecer el referencial de la base.* Es un sistema de coordenadas ortonormal dextrógiro \mathbf{L}_0 en la base soporte, con el eje z_0 a lo largo del eje de movimiento de la primera articulación. Los ejes x_0 e y_0 se pueden establecer convenientemente, para ser normales al eje z_0 . El origen del sistema de coordenadas de referencia es puesto para ser coincidentes con el origen del primer enlace.
2. *Inicializar y repetir.* Para cada articulación k , $k = 1, 2, \dots, n-1$, realizar los pasos 3 al 6.
3. *Establecer los ejes de la articulación.* Alinear el eje z_k con el eje de movimiento (giratorio o deslizante) de la articulación $k+1$.
4. *Establecer el origen del sistema de coordenadas k -ésimo.* Localizar el origen del sistema de coordenadas k -ésimo en la intersección de los ejes z_{k-1} y z_k o en la intersección de las normales de dichos ejes.
5. *Establecer el eje x_k* Establecer $\mathbf{x}_k = +(\mathbf{z}_{k-1} \times \mathbf{z}_k) / \|\mathbf{z}_{k-1} \times \mathbf{z}_k\|$ a lo largo de la normal común entre los ejes z_{k-1} y z_k cuando son paralelos.
6. *Establecer el eje y_k .* Asignar $\mathbf{y}_k = +(\mathbf{z}_k \times \mathbf{x}_k) / \|\mathbf{z}_k \times \mathbf{x}_k\|$ para completar el sistema de coordenadas dextrógiro.

7. *Establecer el sistema de coordenadas de la mano.* Establecer z_n a lo largo, de la dirección del eje z_{n-1} y apuntando hacia fuera del robot. Establecer x_n tal que sea normal a los ejes z_{n-1} y z_n . Asignar y_n para completa el sistema de coordenadas dextrógiro.
8. *Encontrar los parámetros de la articulación y del elemento.* Para $k = 1, \dots, n$, es decir a_k, α_k, d_k y θ_k , determinando las variables de articulación q_k de estos cuatro parámetros.
9. Asociar una matriz de transformación homogénea ${}_{k-1}T^k$ para identificar a cada par articulación-elemento.

2.4 Especificaciones del efector final

Para un robot de n grados de libertad hay n articulaciones, con la articulación k conectando al enlace $k-1$ y al enlace k . Por convención, las articulaciones y los enlaces de un brazo de robot, son enumerados hacia fuera empezando con el enlace 0 fijo en la base y terminando con el enlace n móvil. Para asignar sistemáticamente las coordenadas referenciales a los enlaces de un brazo robótico de n -ejes, la atención especial que se debe poner es en el último enlace, en donde se fija la herramienta o efector final, tal como se muestra en la Fig. (2.6). La posición del efector final se especifica por un vector p respecto del referencial base y la orientación de la herramienta puede ser expresada en coordenadas rectangulares por medio de una matriz de rotación R , donde las tres columnas de esta matriz corresponden respectivamente a los vectores de la normal, de deslizamiento y de aproximación y se expresa mediante una matriz de transformación homogénea como:

$${}^0\mathbf{T}^n = \begin{bmatrix} \mathbf{R} & \mathbf{p} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{n} & \mathbf{s} & \mathbf{a} & \mathbf{p} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.9)$$

- El vector de aproximación \mathbf{a} está alineado con el eje de giro de la herramienta e indica hacia fuera de la muñeca. Este vector especifica la dirección en la cual está apuntando la herramienta.
- El vector de deslizamiento \mathbf{s} , es ortogonal al vector de aproximación y está alineado con el eje de *abre-cierra* de la herramienta.
- El vector normal \mathbf{n} , es ortogonal al plano definido por los vectores de deslizamiento y de aproximación y completa las coordenadas de referencia del sistema dextrógiro.

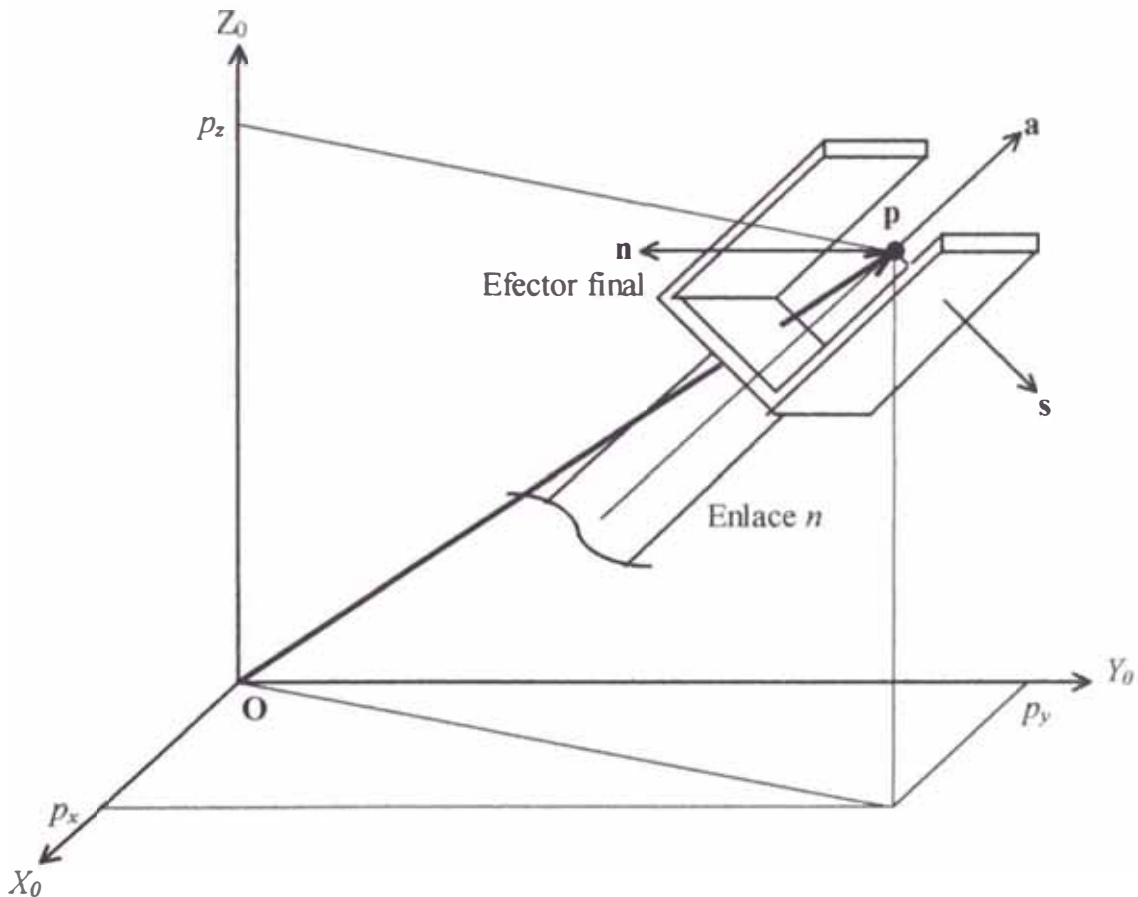


Fig. 2.6 Sistema de coordenadas del efector final.

El efector final se representa por medio de una matriz de transformación homogénea ${}^0T^n$ que describe la posición \mathbf{p} y la orientación \mathbf{R} del efector final con respecto al referencial base.

La matriz \mathbf{R} que viene a ser la representación matricial de la rotación de un cuerpo rígido, está constituido por nueve componentes. Esta matriz simplifica muchas operaciones, pero no nos conduce directamente a un conjunto completo de coordenadas generalizadas; tal conjunto de coordenadas generalizadas se puede obtener mediante los tres ángulos de *Euler* (ϕ , θ y ψ) o los llamados ángulo de giro, de elevación y de desviación respectivamente. En Robótica se pueden usar diferentes configuraciones de los ángulos de *Euler*.

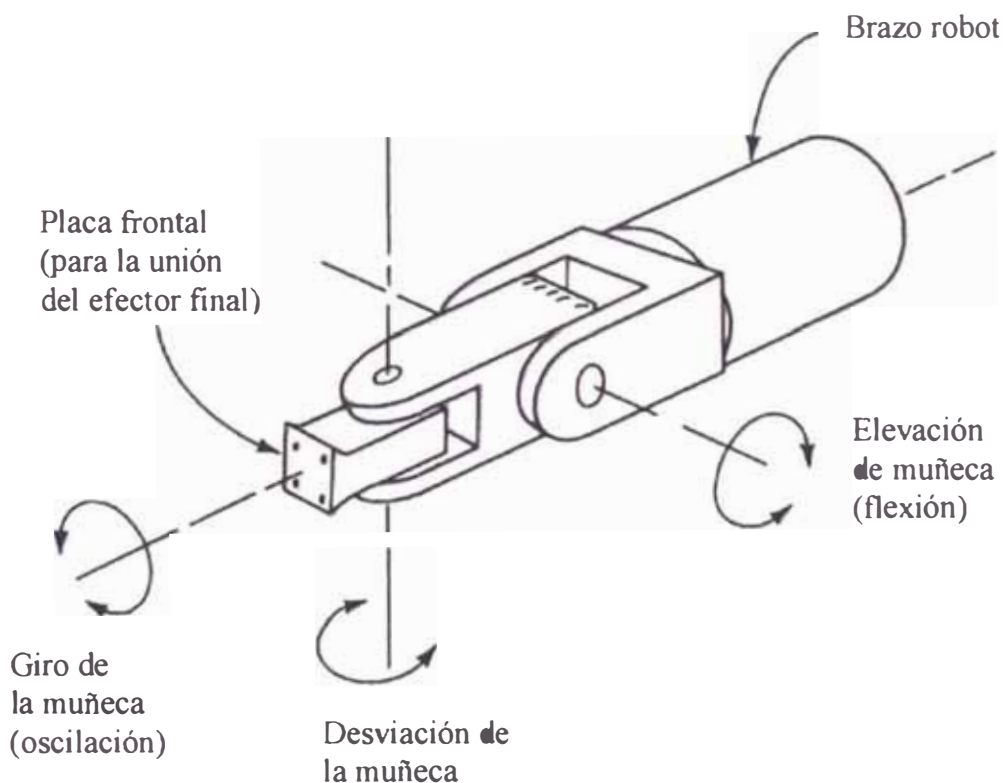


Fig. 2.7 Tres grados de libertad asociados con la muñeca del robot.

El movimiento de la mano o muñeca del robot se diseña para permitir a éste, orientar adecuadamente al efector final o herramienta con respecto a la tarea que debe realizar. Para resolver el problema de la orientación, la mano suele disponer de hasta tres grados de libertad para obtener la configuración de la Fig. (2.7), donde definimos:

- **Giro de la muñeca.** También denominado oscilación de la muñeca, que implica la rotación del mecanismo de la muñeca alrededor del eje del brazo.
- **Elevación de la muñeca.** También denominada flexión de la muñeca, que implica la rotación hacia arriba o abajo de ésta.
- **Desviación de la muñeca.** Implica la rotación a la derecha o a la izquierda de la muñeca.

Utilizando la matriz de rotación con la representación de los ángulos eulerianos como en la ecuación (A.14) del Apéndice A, la matriz de rotación del brazo en el extremo del efector final se puede expresar como:

$${}_{0}\mathbf{R}^n = \begin{bmatrix} C\psi C\theta & C\psi S\theta S\phi - S\psi C\phi & C\psi S\theta C\phi + S\psi S\phi \\ S\psi C\theta & S\psi S\theta S\phi + C\psi C\phi & S\psi S\theta C\phi - C\psi S\phi \\ -S\theta & C\theta S\phi & C\theta C\phi \end{bmatrix} \quad (2.10)$$

Otra ventaja de utilizar la representación de los ángulos de *Euler*, a parte de conducirnos a un sistema de coordenadas generalizadas para la orientación, es que la memoria necesitada en el computador para almacenar la posición y la orientación de un objeto se reduce a un vector de seis elementos $[x \ y \ z \ \phi \ \theta \ \psi]^T$. A partir de este vector, se puede construir la matriz ${}_{0}\mathbf{T}^n$ del brazo del robot mediante la ecuación (2.10) y reemplazándola en la ecuación (2.9).

Como existen diferentes tipos de brazos de robots de acuerdo al movimiento de

sus articulaciones (cartesianas, cilíndricas, esféricas y de brazo articulado) la matriz de transformación del brazo resultante de forma general se expresa por:

$${}^0\mathbf{T}^n = \begin{bmatrix} \mathbf{I}_3 & \mathbf{p}^n \\ 0 & 1 \end{bmatrix} \begin{bmatrix} {}^0\mathbf{R}^n & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^0\mathbf{R}^n & \mathbf{p}^n \\ 0 & 1 \end{bmatrix} \quad (2.11)$$

Así, la orientación ${}^0\mathbf{R}^n$ del efector final, se puede especificar en coordenadas cartesianas o mediante cualquiera de las formas de los ángulos de Euler. La posición \mathbf{p} de la mano $(p_x \ p_y \ p_z)^T$ también se puede especificar en otras coordenadas tales como cilíndricas o esféricas.

De acuerdo a la ecuación (A.25) y a la Fig. (A.9) del Apéndice A, se obtiene la posición en coordenadas cilíndricas de la siguiente manera:

$$\mathbf{p} = \begin{bmatrix} r \cos \alpha \\ r \sin \alpha \\ p_z \end{bmatrix} \quad (2.12)$$

De acuerdo a la ecuación (A.26) y a la Fig. (A.10) del Apéndice A, se obtiene la posición en coordenadas esféricas de la siguiente manera:

$$\mathbf{p} = \begin{bmatrix} r \cos \alpha \sin \beta \\ r \sin \alpha \sin \beta \\ r \cos \beta \end{bmatrix} \quad (2.13)$$

En resumen, existen varios sistemas de coordenadas que se pueden escoger para describir la posición y orientación del efector final. Para el posicionamiento con respecto al referencial base, el vector de posición \mathbf{p} se puede expresar en coordenadas cartesianas, cilíndricas o esféricas. Para describir la orientación \mathbf{R} del efector final con respecto al sistema de coordenadas de la base se tiene la representación cartesiana y los ángulos de Euler en cualquiera de sus modalidades.

2.5 Cinemática directa

La cinemática directa de un robot consiste en expresar y determinar la localización (posición y orientación) del efector final con respecto a un sistema de coordenadas de referencia, conociendo el vector de variables independientes de las articulaciones y los parámetros geométricos de los elementos de la estructura. Para el modelamiento matemático de la cinemática del robot, se usa un método matricial que establece en forma sistemática un sistema de coordenadas (ligado a la estructura) para cada enlace de la cadena poliarticulada, utilizando la representación de *Denavit-Hartenberg*, mediante matrices de transformación homogénea T_k , que relacionan los dos sistemas anteriores, ligados entre sí por un vector de traslación \mathbf{p} , una submatriz de rotación \mathbf{R} , una perspectiva y un escalado, o una combinación de éstas. Para el caso de un robot sólo se usan la traslación y la rotación. La matriz de transformación homogénea representa cada uno de los sistemas de coordenadas de los elementos k , en cada articulación con respecto al sistema de coordenadas del elemento previo $k-1$ con sus respectivos parámetros.

Para expresar la localización del efector final del robot, con respecto al sistema base de coordenadas cartesianas, se multiplican todas las matrices ${}_{k-1}T^k$ correspondientes a cada par elemento-articulación k , obteniendo una matriz de transformación homogénea general ${}_0T^n$ que especifica la localización del n -ésimo referencial con respecto al referencial de la base. Matricialmente se expresa como:

$${}_0T^n = \prod_{k=1}^n {}_{k-1}T^k = {}_0T^1 {}_1T^2 {}_2T^3 \dots {}_{n-1}T^n \quad (2.14)$$

La matriz de transformación ${}_0T^n$ expresa la localización del elemento n que fija al efector final con respecto al sistema de coordenadas cartesianas de referencia de la

base y se puede escribir como:

$${}^0\mathbf{T}^n = {}_{Base}\mathbf{T}^{Efect.\ final} \quad (2.15)$$

Estructuralmente se expresa mediante una rotación y una traslación mediante la siguiente expresión:

$${}^0\mathbf{T}^n = \begin{bmatrix} \mathbf{R} & \mathbf{p} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{n} & \mathbf{s} & \mathbf{a} & \mathbf{p} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.16)$$

donde:

- \mathbf{R} = matriz de orientación 3×3 del elemento n , establecido en el efector final con respecto al sistema de coordenadas de la base.
- \mathbf{p} = vector de posición 3×1 del efector final, que apunta desde el origen del sistema de coordenadas de la base, hasta el origen del sistema de coordenadas, establecido en el punto central de los dedos totalmente cerrados.
- \mathbf{a} = vector de aproximación 3×1 del efector final que apunta en la dirección normal de los dedos de la mano del robot.
- \mathbf{s} = vector de deslizamiento 3×1 del efector final, que apunta en la dirección del movimiento de los dedos cuando la pinza se abre y se cierra.
- \mathbf{n} = vector normal 3×1 del efector final, ortogonal a los dedos de la mano del robot.

Igualando las componentes de cada matriz de la expresión (2.16) se obtiene el vector de variables cartesianas $x(t)$ en función de las variables de articulación como sigue:

$$\mathbf{n} = f_n(q)$$

$$\mathbf{s} = f_s(q)$$

$$\mathbf{a} = f_a(q)$$

$$\mathbf{p} = f_p(q)$$

La solución cinemática directa da una única matriz ${}^0\mathbf{T}^n$ para un vector de articulación \mathbf{q} . Las únicas ligaduras son las acotaciones físicas del vector de articulación \mathbf{q} del robot.

En general la cinemática directa de un robot se define por:

$$\mathbf{x}(t) = f(\mathbf{q}(t)) \quad (2.17)$$

- \mathbf{q} = vector de variables de articulación internas $(q_1 \ q_2 \ q_3 \dots \ q_n)^T$.
- \mathbf{x} = vector de coordenadas de referencia externas $(p_x \ p_y \ p_z \ \psi \ \theta \ \varphi)^T$.
- f = función continua diferenciable en \mathbf{x} con respecto a \mathbf{q} .

Si el manipulador se relaciona a un referencial \mathbf{L}_B mediante una transformación \mathbf{B} y tiene una herramienta unida a la base de montaje en la última articulación descrita por \mathbf{H} , entonces el punto final de la herramienta se relaciona con el sistema de coordenadas de referencia. Multiplicando las matrices \mathbf{B} , ${}^0\mathbf{T}^n$ y \mathbf{H} se obtiene:

$${}_{Ref}\mathbf{T}^{Herram} = \mathbf{B} \ {}^0\mathbf{T}^n \ \mathbf{H} \quad (2.18)$$

2.5.1 Cinemática directa del ROB-DEX

El *ROB-DEX* se representa cinemática y estructuralmente en la Fig. (2.8) como una cadena articulada abierta de cinco elementos rígidos conectados entre sí, por cuatro articulaciones, de las cuales tres son de revolución y la otra es deslizante, las articulaciones están activadas por motores de paso.

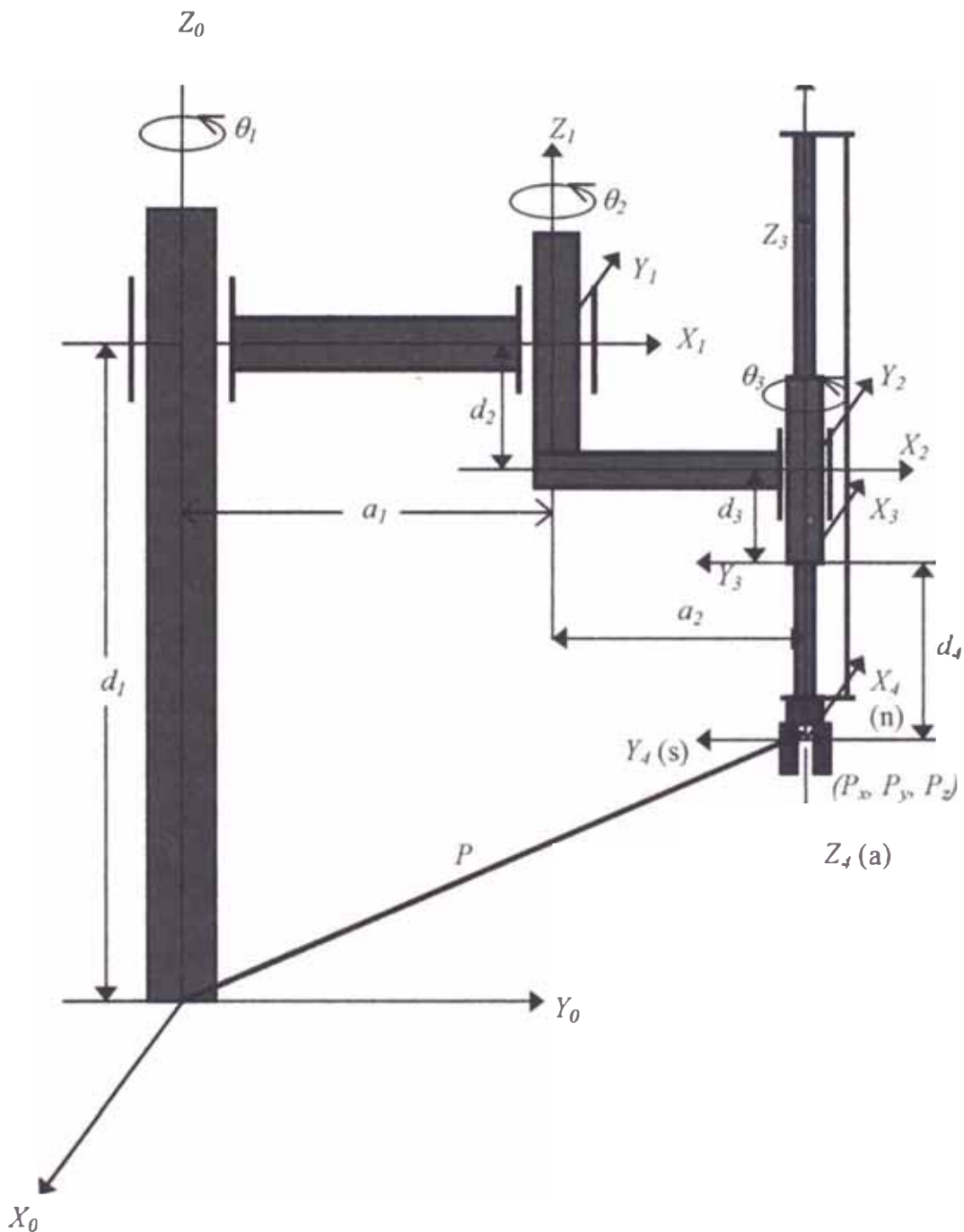


Fig. 2.8 Configuración mecánica del ROB DEX.

El movimiento relativo en las articulaciones resulta en el movimiento de los elementos que posicionan al efector final en una localización deseada. Las dos primeras articulaciones (rotacionales) posicionan y desplazan al efector final a un punto o curva de puntos (p_x, p_y) con respecto al sistema de referencias de coordenadas cartesianas (x_0, y_0, z_0) . La tercera articulación (rotacional), orienta al efector final en

el punto (p_x, p_y) con el eje de orientación en la dirección opuesta al eje z_0 , con respecto del sistema (x_0, y_0) y la cuarta articulación (traslacional), desplaza al efector final a una altura p_z en el espacio cartesiano respecto del eje z_0 del sistema de coordenadas base.

En el extremo del último enlace, se encuentra fijado en el efector final, que viene a ser una pinza de dedos deslizantes que se abre y se cierra en forma paralela.

El espacio o volumen de trabajo en que puede acceder la pinza o efector final del *ROB-DEX* está acotado por:

$$\pi 10^2 p_z \leq \pi (p_x^2 + p_y^2) p_z \leq \pi 60^2 p_z \quad (\text{cm}^3)$$

p_z está limitado por: $0 \leq p_z \leq 31$ (cm).

En la Fig. (2.8) se ha establecido un sistema ortonormal consistente en cada par elemento-articulación y también se muestra la interpretación geométrica de los parámetros de los elementos y articulaciones. Los parámetros y articulaciones del *ROB-DEX* se encuentran en la Tabla (2.1).

| k | α_k | a_k | θ_k | d_k |
|-----|------------|-------|--------------------------------------|------------------------|
| 1 | 0 | 36 cm | $-170^\circ \leq q_1 \leq 170^\circ$ | 75 cm |
| 2 | 0 | 27 cm | $-155^\circ \leq q_2 \leq 155^\circ$ | -12 cm |
| 3 | 0 | 0 cm | $-135^\circ \leq q_3 \leq 135^\circ$ | -10 cm |
| 4 | π | 0 cm | 0° | $-30 \leq -q_4 \leq 0$ |

Tabla 2.1 Parámetros de del *ROB – DEX*.

De los parámetros obtenidos en la Tabla (2.1) se tienen cuatro variables que vienen a ser las articulaciones o grados de libertad del *ROB-DEX*. Tres de éstas son de revolución o angulares q_1, q_2, q_3 y una articulación prismática o deslizante q_4 .

permaneciendo constantes los otros parámetros que caracterizan su estructura física de los elementos.

Como el *ROB-DEX* tiene cuatro grados de libertad, por tanto existen cuatro matrices de transformación homogénea ${}_{k-1}T^k$ asociadas a cada articulación del robot, reemplazando correspondientemente sus valores de sus parámetros de la Tabla (2.1) en la matriz de transformación homogénea de las expresiones (2.3) y (2.7) se obtiene:

$${}^0T^1 = \begin{bmatrix} \cos q_1 & -\text{sen} q_1 & 0 & a_1 \cos q_1 \\ \text{sen} q_1 & \cos q_1 & 0 & a_1 \text{sen} q_1 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.19)$$

$${}^1T^2 = \begin{bmatrix} \cos q_2 & -\text{sen} q_2 & 0 & a_2 \cos q_2 \\ \text{sen} q_2 & \cos q_2 & 0 & a_2 \text{sen} q_2 \\ 0 & 0 & 1 & -d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.20)$$

$${}^2T^3 = \begin{bmatrix} \cos q_3 & -\text{sen} q_3 & 0 & 0 \\ \text{sen} q_3 & \cos q_3 & 0 & 0 \\ 0 & 0 & 1 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.21)$$

$${}^3T^4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & -q_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.22)$$

Para expresar la localización (posición y orientación) del *ROB-DEX* mediante la representación matricial *Denavit-Hartenberg* se aplica directamente la expresión (2.14). Es decir se multiplican ordenadamente las expresiones (2.19), (2.20), (2.21) y (2.22) para obtener:

$$\begin{aligned}
{}_0\mathbf{T}^4 &= {}_0\mathbf{T}^1 {}_1\mathbf{T}^2 {}_2\mathbf{T}^3 {}_3\mathbf{T}^4 {}_0\mathbf{T}^4 \\
\mathbf{T}^4 &= \begin{bmatrix} \cos(q_1 + q_2 + q_3) & \text{sen}(q_1 + q_2 + q_3) & 0 & a_1 \cos(q_1) + a_2 \cos(q_1 + q_2) \\ \text{sen}(q_1 + q_2 + q_3) & -\cos(q_1 + q_2 + q_3) & 0 & a_1 \text{sen}(q_1) + a_2 \text{sen}(q_1 + q_2) \\ 0 & 0 & -1 & d_1 - d_2 - d_3 - q_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned} \tag{2.23}$$

Para expresar la posición y orientación del efector final con respecto al sistema de referencia base $(x_0 \ y_0 \ z_0)$ se hace uso de la expresión (2.16) y se obtiene:

$$\begin{aligned}
{}_0\mathbf{T}^4 &= {}_{base}\mathbf{T}^{herram} \\
\begin{bmatrix} C_{123} & S_{123} & 0 & a_1 C_1 + a_2 C_{12} \\ S_{123} & -C_{123} & 0 & a_1 S_1 + a_2 S_{12} \\ 0 & 0 & -1 & d_1 - d_2 - d_3 - q_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} &= \begin{bmatrix} \cos \varphi & \text{sen} \varphi & 0 & p_x \\ \text{sen} \varphi & -\cos \varphi & 0 & p_y \\ 0 & 0 & -1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned} \tag{2.24}$$

donde:

$$C_{123} = \cos(q_1 + q_2 + q_3), \quad S_{123} = \text{sen}(q_1 + q_2 + q_3).$$

- $(p_x \ p_y \ p_z)$ describe la posición espacial del efector final con respecto al sistema base de coordenadas $(x_0 \ y_0 \ z_0)$.
- φ representa el ángulo de orientación del efector final con respecto al eje z_0 .

Igualando los términos de las matrices expresadas por las ecuaciones (2.23) y (2.24) se obtiene la cinemática directa como:

$$p_x = a_1 \cos q_1 + a_2 \cos(q_1 + q_2) \tag{2.25.a}$$

$$p_y = a_1 \text{sen} q_1 + a_2 \text{sen}(q_1 + q_2) \tag{2.25.b}$$

$$p_z = d_1 - d_2 - d_3 - q_4 \tag{2.25.c}$$

$$\varphi = q_1 + q_2 + q_3 \tag{2.25.d}$$

De las ecuaciones (2.25) se puede expresar la cinemática directa del *ROB-DEX* como una función en el tiempo, representada por la ecuación (2.17):

$$x(t) = f(q(t)) \quad (2.26)$$

- q = vector de variables de articulación internas (q_1 q_2 q_3 q_4).
- x = vector de coordenadas de referencia externas (p_x p_y p_z φ).

2.6 Cinemática inversa

La cinemática inversa de un robot o solución del brazo para un manipulador con n grados de libertad, consiste en determinar las magnitudes del vector de variables de articulación con el fin de controlar la posición y orientación del efector final para alcanzar su objetivo en la localización deseada; es decir conociendo la localización (posición y orientación) del efector final del robot de n grados de libertad y los parámetros geométricos de los elementos de su estructura; nos gustaría encontrar los ángulos de articulación $q(t)$ del robot de manera que se pueda posicionar como se desee al efector final y determinar además si el efector final del robot puede o no alcanzar dicha localización y de cuántas maneras, debido que la solución de la cinemática inversa de un robot no es única.

Es deseable encontrar una solución del brazo en forma cerrada para los manipuladores. La mayoría de los robots comerciales tienen una u otra forma de las condiciones siguientes que hacen posible la solución del brazo en forma cerrada:

1. Tres ejes de articulación adyacentes se interceptan en un punto.
2. Tres ejes de articulación adyacentes son paralelos entre sí.

El *ROB-DEX* satisface la segunda condición para poder encontrar la solución en

forma cerrada.

La cinemática inversa de un robot, se expresa como una función del tiempo usando la expresión (2.17) mediante la siguiente función inversa:

$$q(t) = f^{-1}(x(t)) \quad (2.27)$$

Para evaluar la cinemática inversa de un robot existen diversos métodos: analíticos, geométricos e iterativos. El método seleccionado depende de la configuración geométrica del robot, del tipo de trayectoria que se quiera usar o de la aplicación para lo cual está diseñado. A continuación se describen tales métodos:

- a) **La solución analítica**, emplea las matrices de transformación inversa de cada elemento y articulación. Este método es literalmente exacto y muy útil cuando se tenga una trayectoria punto a punto; sin embargo, se presentan puntos de singularidad cuando existe redundancia del robot y otras veces se obtienen soluciones finitas múltiples creando una incertidumbre en la solución.
- b) **La solución numérica** o iterativa a menudo requiere más cálculo y no garantiza la convergencia a la solución correcta, especialmente en los casos singulares y degenerados. Este método se basa en la linealización de las ecuaciones de la transformada directa, empleando el algoritmo multidimensional del tipo *Newton-Rapson*. Este método es ventajoso por su universalidad, evita singularidades, su convergencia es rápida y es apropiado cuando se sigue una trayectoria controlada; sin embargo, es problemático para la determinación de los valores iniciales y del tiempo de computación empleado en cada iteración.

Para determinar la cinemática inversa del robot por el método analítico se hacen uso de las ecuaciones de la cinemática directa y de determinan los valores de las articulaciones q en función de las coordenadas de referencia del efector final y de sus

parámetros físicos. Normalmente este tipo de solución se realiza utilizando matrices inversas de transformación homogénea.

Para evaluar la cinemática inversa del robot usando el método iterativo o numérico, primero se debe linealizar el sistema de ecuaciones (2.17) mediante la serie de *Taylor* dada a continuación:

$$x(q) = x(\bar{q} + \Delta q) = x(\bar{q}) + \frac{\partial x(\bar{q})}{\partial q}(q - \bar{q}) + \frac{\partial^2 x(\bar{q})}{2! \partial q^2}(q - \bar{q})^2 \dots \quad (2.28)$$

Considerando solamente el término lineal de la expresión (2.28) por el método de *Newton-Rapson* para hallar la solución de un sistema de m ecuaciones con n variables, tendremos:

$$x(q) - x(\bar{q}) = \frac{\partial x(\bar{q})}{\partial q}(q - \bar{q}) \quad (2.29)$$

El término de la derecha de la ecuación (2.29) representa la *matriz Jacobiana*, la cual se puede expresar como sigue:

$$J(q) = \frac{\partial x(\bar{q})}{\partial q} = \begin{bmatrix} \frac{\partial}{\partial q_1} x_1(\bar{q}) & \frac{\partial}{\partial q_2} x_1(\bar{q}) & \dots & \frac{\partial}{\partial q_{n-1}} x_1(\bar{q}) & \frac{\partial}{\partial q_n} x_1(\bar{q}) \\ \vdots & \vdots & \dots & \dots & \vdots \\ \frac{\partial}{\partial q_1} x_m(\bar{q}) & \frac{\partial}{\partial q_2} x_m(\bar{q}) & \dots & \frac{\partial}{\partial q_{n-1}} x_m(\bar{q}) & \frac{\partial}{\partial q_n} x_m(\bar{q}) \end{bmatrix} \quad (2.30)$$

Si la matriz $J(q)$ es no singular y además $m = n$, entonces se puede despejar el vector q de la ecuación (2.29) y luego de realizar operaciones elementales se llega a una expresión que toma la forma de una función recursiva e iterativa conocidos los vectores: articular y cartesiano anteriores, obteniéndose por lo tanto:

$$q_{k+1} = q_k + J^{-1}(q)[x_{k+1} - x_k] \quad (2.31)$$

2.6.1 Cinemática inversa del ROB-DEX

Para determinar la cinemática inversa del *ROB-DEX* se emplean los dos métodos indicados anteriormente: analítico y numérico, dependiendo de las aplicaciones que esté realizando el robot y del tipo de trayectoria que esté siguiendo.

Para evaluar la cinemática inversa del *ROB-DEX*, por el método analítico se usan las expresiones no lineales trigonométricas de la cinemática directa, obteniendo valores literalmente exactos con ausencia de errores acumulados, y, cuando se deriven sus expresiones, las singularidades del manipulador son fácilmente detectados. Utilizando las ecuaciones (2.25.a) y (2.25.b) y operando adecuadamente se obtiene:

$$p_x \cos q_1 + p_y \sin q_1 = \frac{p_x^2 + p_y^2 + a_1^2 - a_2^2}{2a_1} = Z \quad (2.32)$$

Usando las ecuaciones (2.25) y la ecuación (2.32) se obtiene la solución de la cinemática inversa para el *ROB-DEX* en forma explícita :

$$q_1 = \arctan \left(\frac{p_y}{p_x} \right) + \arctan \left(\frac{\pm \sqrt{p_x^2 + p_y^2 - Z^2}}{Z} \right) \quad (2.33.a)$$

$$q_2 = \arctan \left(\frac{p_y - a_1 \sin q_1}{p_x - a_1 \cos q_1} \right) - q_1 \quad (2.33.b)$$

$$q_3 = \varphi - q_1 - q_2 \quad (2.33.c)$$

$$q_4 = d_1 - d_2 - d_3 - p_z \quad (2.33.d)$$

El doble signo de la expresión (2.33.a) indica que existen dos soluciones que posicionan al efector final del robot en un mismo punto. Para el *ROB-DEX*, el signo (+) significa que está usando la configuración de brazo derecho y el signo (-) que está

usando la configuración de brazo izquierdo.

Para evaluar la cinemática inversa del *ROB-DEX* usando el método iterativo se usa la linealización; empleando la serie de *Taylor* y aplicando el método de *Newton-Rapson* haciendo uso de la ecuación (2.31), donde la matriz Jacobiana J de 4×4 cuadrada es no singular y se obtiene usando la expresión (2.30), entonces tal matriz Jacobiana del *ROB-DEX* es como sigue:

$$J = \begin{bmatrix} -a_1 S_1 - a_2 S_{12} & -S_{12} & 0 & 0 \\ a_1 C_1 + a_2 C_{12} & C_{12} & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \quad (2.34)$$

2.7 Singularidades

Para transformar las coordenadas cartesianas deseadas en consignas articulares para el controlador del robot, es necesario contar con el modelo cinemático inverso. Del mismo modo, para encontrar las velocidades articulares necesarias para seguir una trayectoria definida por las velocidades cartesianas generalizadas del efector final, será necesario contar con el modelo variacional del robot. Dicho modelo se obtiene derivando e invirtiendo la ecuación (2.26), y como resultado se obtiene:

$$\dot{q} = J^{-1}(q) \dot{x} \quad (2.35)$$

Entonces, para poder resolver el problema de cinemática variacional inversa, se necesita que el Jacobiano no tenga singularidades que impidan su inversión. Muchos robots tiene valores articulares q para los cuales la matriz Jacobiana expresada por la ecuación (2.30) es singular; estos casos se denominan *configuraciones singulares*. Cuando un robot se encuentra en una configuración singular, pierde uno o más

grados de libertad. Esto ocurre generalmente en las fronteras del espacio de trabajo, pues ahí está presente la posibilidad de reconfigurar su vector de coordenadas articulares. Al interior del espacio de trabajo también puede haber singularidades, lo cual ocurre cuando dos o más ejes articulares están alineados, haciendo indistintas las variables articulares asociadas.

En las configuraciones singulares existe un subespacio cartesiano de direcciones a lo largo de las cuales es imposible mover el efector del robot, independientemente de los desplazamientos articulares diferenciales (o velocidades) especificados. Obviamente esto siempre sucede en las fronteras del espacio de trabajo de todo robot.

CAPITULO III

PLANIFICACION Y CONTROL DE TRAYECTORIAS DEL ROBOT

3.1 Introducción

En el diseño de estaciones de trabajo robotizadas, típicamente los temas que se estudian conciernen al esquema de componentes de la estación, la morfología del robot manipulador, la localización relativa del robot y la geometría de la trayectoria del efector final del robot. Algunos problemas de factibilidad de tareas relacionadas con los temas de diseño de estaciones robotizadas de trabajo, se relacionan con la presencia o no de obstáculos en las zonas accesibles al efector final del robot, ya que éste podría colisionar con tales obstáculos presentes en la estación de trabajo.

La cinemática de un brazo robótico proporciona un modelo matemático que puede ser usado para colocar la herramienta o efector final en una posición y orientación arbitraria dentro del espacio de trabajo. Usando las formulaciones de cinemática como un paso previo, un problema de más alto nivel viene a ser la planificación de trayectorias a seguir por las herramientas, para desempeñar tareas significativas de manipulación.

Para formular adecuadamente el problema de planificación de tareas, se debe comenzar con un examen detallado del espacio de trabajo del robot. Seguidamente se discutirán técnicas sofisticadas de planificación de trayectorias que incluye trayectoria continua de movimiento mediante el uso de la cinemática inversa. Los

métodos más avanzados han resultado ser útiles para aplicaciones tal como pintando a spray, soldadura por arco, engomando, etc.

3.2 Análisis del espacio de trabajo

El espacio o volumen de trabajo se basa en el conjunto de puntos en \mathbf{R}^3 que puede ser alcanzado por la muñeca del robot. Analizando el ámbito de trabajo basado en las posiciones alcanzables por el extremo de la herramienta como un punto de referencia, se debe incluir los efectos tanto de los ejes mayores usados para ubicar la muñeca como de los ejes menores usados para orientar la herramienta. La forma o geometría del ámbito de trabajo varía de robot a robot; sin embargo, el ámbito de trabajo puede también analizarse dentro de la estructura de espacio articular de \mathbf{R}^n . En el espacio de articulación, el ámbito de trabajo se caracteriza típicamente por los límites resultantes de la combinación lineal de las variables de articulación. Las limitaciones de esta naturaleza generan un poliedro convexo en \mathbf{R}^n , denominado *ámbito de trabajo del espacio de articulación*.

Si \mathbf{q}_{\min} y $\mathbf{q}_{\max} \in \mathbf{R}^n$ son vectores que denotan los límites de las articulaciones y \mathbf{C} es una matriz de acoplamiento de $m \times n$, entonces el conjunto de todos los valores que las variables de articulación q pueden asumir se llama *ámbito de trabajo del espacio de articulación accesible*, denotado por \mathbf{Q}_a y definido como:

$$\mathbf{Q}_a \equiv \{ \mathbf{q} \in \mathbf{R}^n : \mathbf{q}_{\min} \leq \mathbf{q} \leq \mathbf{q}_{\max} \} \quad (3.1)$$

donde:

\mathbf{q}_{\min} es el límite mínimo del vector de articulación q y \mathbf{q}_{\max} es el límite máximo del vector de articulación q .

El caso especial más simple de ámbito de trabajo del espacio de articulación ocurre cuando $\mathbf{C} = \mathbf{I}$, lo cual no representa acoplamiento entre ejes. En este caso, el número de limitaciones m equipara al número de variables de articulación n . En general, la matriz de acoplamiento de articulación \mathbf{C} puede ser rectangular ($m \neq n$) y puede contener términos diagonales diferentes de cero. Esto ocurre por ejemplo, cuando hay límites en combinaciones lineales (sumas o diferencias) de variables de articulación.

El ámbito de trabajo del espacio de articulación accesible \mathbf{Q}_a es relativamente fácil de caracterizar; éste es el conjunto de puntos en \mathbf{R}^3 que puede ser alcanzado por el extremo de la herramienta que es lo que interesa. Hay diferentes tipos de ámbitos de trabajo en \mathbf{R}^3 . Por ejemplo, el conjunto de todos los puntos $p \in \mathbf{R}^3$ alcanzables desde una orientación específica por el extremo de la herramienta se refiere al ámbito de trabajo total, o simplemente ámbito de trabajo, y se denota por \mathbf{Y} . Alternativamente, el conjunto de los puntos alcanzables desde una orientación arbitraria de la herramienta se llama ámbito de trabajo diestro y se denota por \mathbf{Y}_d . El ámbito de trabajo diestro es más pequeño pero más práctico que el ámbito de trabajo total. El ámbito de trabajo de un manipulador puede ser expresado directamente en términos del ámbito de trabajo del espacio de articulación accesible \mathbf{Q}_a y en función del extremo de la posición $p(q)$ como sigue:

$$\mathbf{Y} = \{p(q) \in \mathbf{R}^3 : q \in \mathbf{Q}_a\} \quad (3.2)$$

El ámbito de trabajo \mathbf{Y} expresado matemáticamente, es la imagen en el ámbito de trabajo del espacio de articulación \mathbf{Q}_a bajo la función de la posición de herramienta $p(q)$, que corresponde a la cuarta columna de la matriz ${}_{Base} \mathbf{T}^{Efec. final}(q)$ del brazo.

Alternativamente, la posición y orientación del efector final de un robot manipulador son descritos por un vector $x \in \mathbf{R}^m$. La función de la posición $p(q)$ de la herramienta o efector final consiste de las tres primeras componentes del vector $x(q)$ de la configuración de la herramienta. Las m componentes de x son llamadas coordenadas operacionales del efector final del robot ($m \leq 6$). El vector x se puede escribir de la siguiente forma:

$$\mathbf{x} = [x_1 \quad \cdots \quad x_m]^T \quad (3.3)$$

La configuración articular de un robot manipulador está definida por un vector $\mathbf{q} \in \mathbf{R}^n$. Las n componentes de \mathbf{q} son llamadas coordenadas generalizadas del robot, que describen la posición relativa de un eslabón del robot, respecto del eslabón que le precede. El vector \mathbf{q} se puede escribir de la siguiente forma:

$$\mathbf{q} = [q_1 \quad q_2 \quad \cdots \quad q_n]^T \quad (3.4)$$

El conjunto de configuraciones libre de colisiones de un robot, es el conjunto \mathbf{Q}_f queda expresado como:

$$\mathbf{Q}_f \equiv \{q \in \mathbf{Q}_a \mid M(q) \cap E_0 = \emptyset\} \quad (3.5)$$

donde $M(q)$ es la superficie del espacio cartesiano ocupado por los eslabones del robot y E_0 es la superficie del espacio cartesiano ocupado por los obstáculos en el ámbito de trabajo.

El espacio de trabajo libre W_f es la imagen en el espacio operacional del conjunto de configuraciones libre de colisiones \mathbf{Q}_f , bajo el modelo de la posición cinemática directa o sea:

$$W_f = f(\mathbf{Q}_f) \quad (3.6)$$

3.3 Planificación de la trayectoria

Habiendo resuelto el problema de la cinemática directa e inversa, el problema primario que surge en el control de un robot, es la generación del movimiento del manipulador en ausencia de perturbaciones respecto de la trayectoria nominal.

Para controlar un manipulador se requiere que la configuración de su efector final, tanto en la posición inicial como en la final se especifique, antes de que se planifique la trayectoria del movimiento.

Esta sección dedica su atención a diversos sistemas de planificación de trayectorias para movimientos libres de obstáculos. También trata con el formalismo de describir el movimiento preplanificado del brazo del robot como una secuencia de puntos en el espacio (posición y orientación) a través de los cuales debe pasar el manipulador. Así mismo, se ocupa de la curva en el espacio que recorre desde una posición inicial de localización hasta una posición final. La planificación de trayectorias se puede realizar o bien en el espacio de las variables de articulación \mathbf{R}^n o en el espacio cartesiano \mathbf{R}^m .

La propiedad básica con la cual los algoritmos de generación de movimiento son abastecidos por su funcionalidad debe proporcionar continuidad y uniformidad al movimiento entre las posiciones y orientaciones del efector final, especificadas por el usuario. Junto a este requerimiento básico, propiedades adicionales son también necesarias para incrementar la adaptabilidad y flexibilidad de los robots. Por ejemplo, los algoritmos de generación de movimiento podrían proporcionar formas para eludir obstáculos en el empleo de sensores tales como cámaras, sensores táctiles, sensores de fuerza, sensores de proximidad, etc.

La generación del movimiento de un manipulador usualmente requiere:

1. La especificación de un conjunto de puntos (nudos) en el espacio de coordenadas cartesianas o en coordenadas de articulación por los cuales el efector final del manipulador debería pasar.
2. Definición y evaluación de la trayectoria entre estos puntos.

Los esquemas de planeamiento y control de trayectorias generalmente "interpolan" o "aproximan" la trayectoria programada mediante una clase de funciones polinomiales y generan una secuencia de "puntos de consigna para control" a lo largo del tiempo, para controlar al robot desde su posición de origen hasta su destino. Los extremos del camino se pueden especificar en coordenadas de articulación generado por el vector de variables $q(t)$ o bien en coordenadas de referencia cartesianas $x(t)$ para mover al robot según la tarea a realizar. No obstante, se suelen especificar en coordenadas cartesianas, debido que es más fácil visualizar las configuraciones correctas del efector final. Mas aún, las coordenadas de articulación no son adecuadas como un sistema de coordenadas de trabajo, porque los ejes de las articulaciones de la mayoría de los robots no son ortogonales y no distinguen la orientación con la posición.

3.4 Clasificación de las trayectorias

Existen una serie de trayectorias posibles para realizar entre dos puntos extremos específicos, por los que debe pasar el efector final del robot manipulador para realizar alguna tarea. Las trayectorias, ya sea controladas respecto de las variables articulares o respecto del espacio cartesiano se pueden clasificar en: trayectorias punto a punto, trayectorias continuas, trayectorias continuas controladas y trayectorias optimizadas.

- **Trayectorias punto a punto**

En este tipo de trayectorias movemos al efector final del robot desde un punto inicial hasta un punto destino sin importar cómo evoluciona el efector final en el camino que está siguiendo. La definición de la precisión de posicionamiento en el punto final puede ser programada en un valor mínimo y en un valor máximo dependiendo de las características del robot. Este método requiere por parte del computador, menor tiempo de cálculo de los algoritmos.

El *ROB-DEX* usa este tipo de trayectoria en aplicaciones que necesiten precisión sólo en los puntos de partida y de llegada sin importar la forma como lo hace; es decir independiente del camino realizado.

- **Trayectorias continuas**

En este tipo de trayectoria, el efector final o herramienta del robot se mueve a lo largo de una trayectoria definida normalmente a través de interpolaciones lineales en el dominio cartesiano y en función del tiempo. Para esto es necesario generar múltiples puntos intermedios en la curva que describe la trayectoria, invirtiendo mayor tiempo de cálculo en evaluar la cinemática inversa con el fin de obtener una buena precisión en el desplazamiento. Al inicio y final del camino y durante el movimiento, se pueden presentar grandes aceleraciones y velocidades imprevistas. En este tipo de trayectoria lo más adecuado es aplicar el método numérico de la cinemática inversa.

El *ROB-DEX* usa esta clase de trayectorias en aplicaciones donde se requiere que el efector final se desplace mediante una curva programada de puntos con una precisión aceptable. Aquí no se toma en cuenta la forma de la velocidad ni de la aceleración.

- **Trayectorias continuas controladas**

En este tipo de trayectoria se programa y controla el movimiento, velocidad y aceleración como funciones continuas y derivables del tiempo, así como también se pueden imponer restricciones de su realizabilidad física. La forma de generar este tipo de trayectorias es a través de interpolaciones polinómicas combinadas de segundo, tercer, cuarto y grado superior en función del tiempo, bien en el dominio cartesiano o bien en el dominio de articulación, estableciéndose condiciones de frontera, ligaduras de trayectoria y condiciones iniciales; pero con un elevado costo computacional.

El *ROB-DEX* utiliza este tipo de trayectoria en su mayoría de acciones, controlando su posición, su velocidad y en algunos caso su aceleración, con respecto al espacio de articulación y con respecto al espacio cartesiano.

- **Trayectorias optimizadas**

En este tipo de trayectorias se usan aparte de los criterios anteriores, otros parámetros físicos que tienen como fin esencial:

- Disminuir el consumo energético del robot.
Mejorar las rampas de aceleración y frenado de los ejes.
- Disminuir el desgaste de las partes electromecánicas.
Optimizar las fuerzas y torques empleados.
- Aumentar el rendimiento y eficacia de los medios de producción.

Para esto es necesario contar con métodos y algoritmos numéricos programados que trabajen sincrónicamente con los interpoladores de velocidad y posicionamiento en forma adelantada.

3.5 Generación de trayectorias articulares

Para la planificación de las variables de articulación, se planifica la historia temporal de todas las variables de articulación y de sus dos primeras derivadas respecto del tiempo, a fin de describir al movimiento deseado del manipulador.

La planificación en el espacio de las variables de articulación tiene tres ventajas:

1. La trayectoria se planifica directamente en términos de las variables controladas durante el movimiento.
2. La planificación de trayectoria se puede hacer casi en tiempo real.
3. Las trayectorias de la articulación son más fáciles de planificar.

La desventaja asociada es la dificultad en determinar las posiciones de los diversos elementos y del efector final durante el movimiento, condición que se suele necesitar para garantizar evitar obstáculos a lo largo de la trayectoria respecto del sistema de referencia.

Este tipo de interpolación generalmente se usa cuando no se requiere el recorrido del efector final por una determinada trayectoria. Entre los tipos de interpolaciones más comunes tenemos: interpolación polinómica lineal, cúbica y de grado superior.

3.5.1 Interpolación polinómica lineal

El caso más simple de interpolación en el espacio de la configuración articular de la herramienta en un intervalo de tiempo $[t_0, t_f]$, es la utilización de un polinomio lineal para cada articulación de un robot de n grados de libertad. Mediante esta interpolación se desplaza la articulación variando desde un punto inicial q_0 hasta un punto final q_f . Para determinar una trayectoria interpolada es conveniente usar una variable de tiempo normalizado τ , que permite tratar a la trayectoria de cada

articulación de la misma manera, con el tiempo real t variando desde t_0 hasta t_f . La variable de tiempo normalizada $\tau(t)$ se define como:

$$\tau(t) = \frac{t - t_0}{t_f - t_0} = \frac{t - t_0}{T} \quad (3.7)$$

$$\tau_0 = 0 ; \quad \tau_f = 1$$

La trayectoria para cada articulación consiste en definir un polinomio de primer orden $q(t)$ en función τ como:

$$q(t) = q_0 + \tau(t)(q_f - q_0) \quad (3.8)$$

La expresión de la velocidad $w(t)$ se obtiene derivando la ecuación (3.8):

$$w(t) = \frac{dq(t)}{dt} = \frac{d\tau(t)}{dt}(q_f - q_0) = \frac{(q_f - q_0)}{T} \quad (3.9)$$

donde:

τ : tiempo normalizado, $\tau \in [0, 1]$.

t : tiempo real en segundos, $t \in [t_0, t_f]$

t_0 : tiempo real al inicio de la trayectoria.

t_f : tiempo real al final de la trayectoria.

$T = t_f - t_0$: tiempo real necesitado para recorrer toda la trayectoria.

$q(t)$: vector de variables articulares $q(t) \in \mathbf{R}^n$.

q_0 : posición inicial del vector de la articulación.

q_f : posición final del vector de la articulación.

$w(t)$ = vector de velocidad de articulación

De la expresión (3.8) se evalúa el movimiento de la articulación, el cual varía linealmente en función del tiempo a una velocidad constante tal como se muestra en

la Fig. 3.1, lo que indica que no existe aceleración en el intervalo $\langle t_0, t_f \rangle$. Este tipo de trayectoria presenta una discontinuidad de la velocidad en los puntos inicial y final, generando vibraciones y sacudidas en su estructura mecánica, disminuyendo la precisión en estos puntos y perjudicando a los actuadores de las articulaciones, sobre todo cuando el robot se mueve a grandes velocidades.

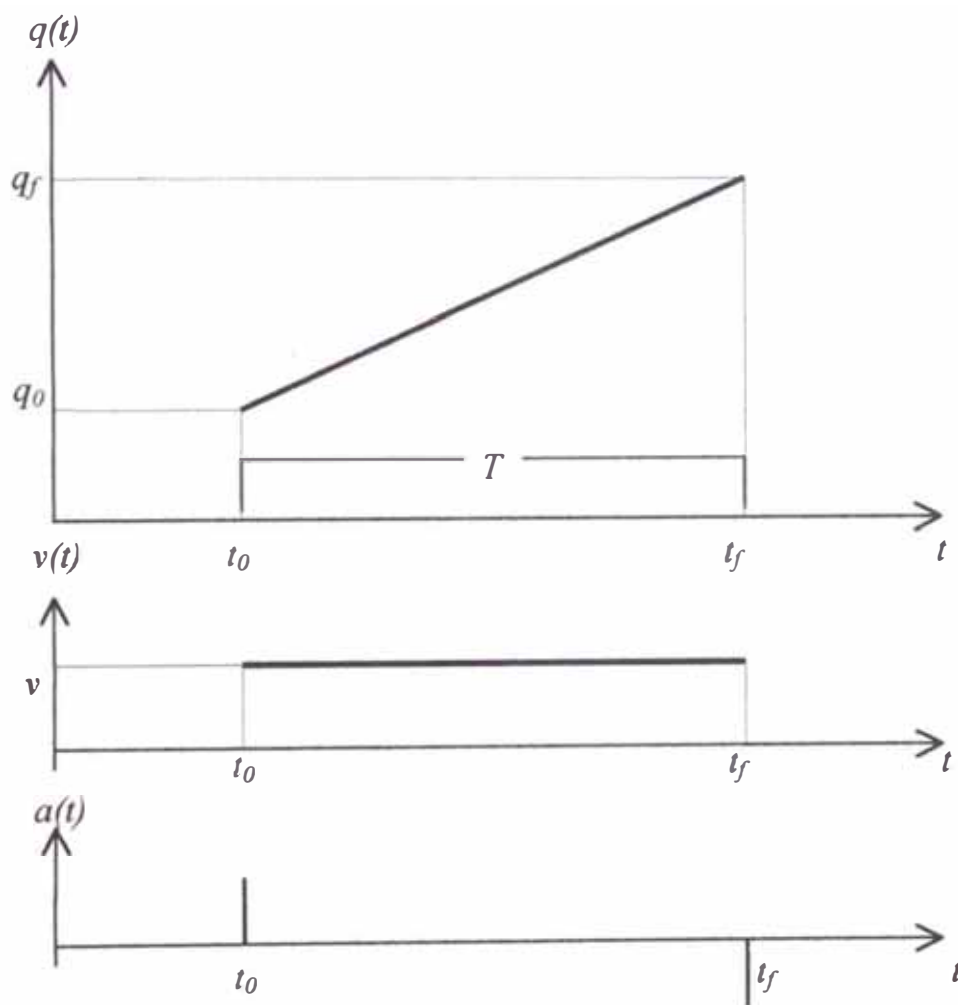


Fig. 3.1 Interpolación lineal: posición, velocidad y aceleración.

La gran ventaja de usar este tipo de trayectoria radica en su menor costo computacional para efectos de cálculo sobre todo cuando se requiere dar aprendizaje al robot.

3.5.2 Interpolación polinómica cúbica

La trayectoria $q(t)$ de las articulaciones entre dos puntos específicas debería ser suave; entonces, $q(t)$ debe tener al menos dos derivadas continuas a fin de evitar aceleraciones infinitas. En la práctica, en la presencia de aceleraciones altas, el movimiento del robot se manifiesta inestable, y puede deteriorar los circuitos del actuador o saturar a los actuadores. Se hará el análisis del movimiento interpolado, cuando el robot se mueve desde un punto q_0 a un punto q_f , en el espacio de la configuración articular en un intervalo de tiempo $[t_0, t_f]$ entre los puntos de una trayectoria cúbica polinomial suave.

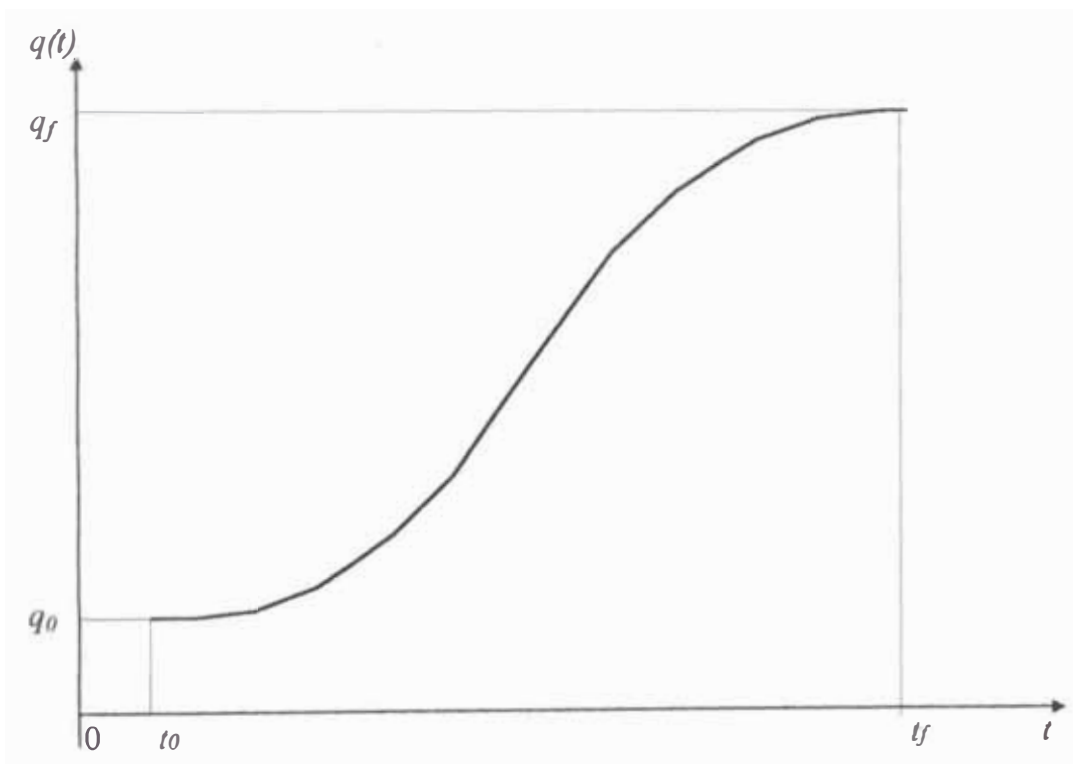


Fig. 3.2 Interpolación posicional cúbica.

El polinomio que representa una trayectoria de tercer grado en el espacio de variables de articulación del robot en función del tiempo normalizado τ , se representa en la Fig. (3.2) y se expresa mediante la siguiente expresión polinómica:

$$p(t) = p(\tau(t)) = a_3 \tau^3(t) + a_2 \tau^2(t) + a_1 \tau(t) + a_0 \quad (3.10)$$

La velocidad $w(t)$ de la articulación o primera derivada de $p(t)$ en función de τ está dada por la siguiente expresión:

$$w(\tau(t)) = \frac{dp(\tau)}{d\tau} \frac{d\tau}{dt} = \frac{1}{T} \frac{dp(\tau)}{d\tau} = \frac{1}{T} (3a_3 \tau^2 + 2a_2 \tau + a_1) \quad (3.11)$$

La aceleración $\alpha(t)$ de la articulación o segunda derivada de $p(t)$ en función de τ , se expresa de la siguiente manera:

$$\alpha(\tau(t)) = \frac{d^2 p(\tau)}{d\tau^2} \left(\frac{d\tau}{dt}\right)^2 = \frac{1}{T^2} \frac{d^2 p(\tau)}{d\tau^2} = \frac{1}{T^2} (6a_3 \tau + 2a_2) \quad (3.12)$$

Las limitaciones o ligaduras de posición en los puntos inicial y final de la trayectoria respectivamente son $p(t_0) = q_0$ y $p(t_f) = q_f$ y las ligaduras de velocidad en estos puntos respectivamente son $w(t_0) = w_0$ y $w(t_f) = w_f$.

Al evaluar las expresiones (3.10) y (3.11) usando las ligaduras en el punto inicial se tiene:

$$q_0 = p(0) = a_0 \quad (3.13)$$

$$w_0 = w(0) = \frac{a_1}{T} \quad (3.14)$$

Al evaluar las expresiones (3.10) y (3.11) usando las ligaduras en el punto final se tiene:

$$q_f = p(1) = a_3 + a_2 + a_1 + a_0 \quad (3.15)$$

$$w_f = w(1) = \frac{3a_3 + 2a_2 + a_1}{T} \quad (3.16)$$

De las expresiones (3.13), (3.14), (3.15) y (3.16) podemos obtener los coeficientes del polinomio $p(t)$ como sigue:

$$a_0 = q_0 \quad (3.17.a)$$

$$a_1 = v_0 T \quad (3.17.b)$$

$$a_2 = 3(q_f - q_0) - (2w_0 + w_f)T \quad (3.17.c)$$

$$a_3 = -2(q_f - q_0) + (w_0 + w_f)T \quad (3.17.d)$$

Si definimos w_m como la velocidad media entre los puntos inicial y final durante el tiempo T como:

$$w_m = \frac{q_f - q_0}{T} \quad (3.18)$$

Reemplazando los valores de los coeficientes a_i y usando w_m en las ecuaciones (3.10), (3.11) y (3.12) se obtiene la expresión del desplazamiento, velocidad y aceleración respectivamente como:

$$q(\tau) = [(w_f + w_0)T - 2(q_f - q_0)]\tau^3 - [(2w_0 + w_f)T - 3(q_f - q_0)]\tau^2 + w_0 T \tau + q_0 \quad (3.19)$$

$$w(\tau) = 3 [(w_f + w_0) - 2w_m]\tau^2 + 2 [3w_m - (2w_0 + w_f)]\tau + w_0 \quad (3.20)$$

$$\alpha(\tau) = \frac{1}{T} \left\{ 6 [(w_0 + w_f) - 2w_m]\tau + 2 [3w_m - (2w_0 + w_f)] \right\} \quad (3.21)$$

donde:

- q_0 = vector de articulación en el punto inicial de la trayectoria.
- q_f = vector de articulación en el punto final de la trayectoria.
- w_0 = velocidad inicial de la trayectoria.
- w_f = velocidad final de la trayectoria.
- t = tiempo real que varia entre t_0 y t_f
- T = tiempo de duración del movimiento en la trayectoria.

Para no tener trayectorias indeseadas en el tiempo se puede limitar la velocidad inicial y final de cada articulación mediante la siguiente condición:

$0 \leq w_0, w_f \leq w_m$ Para un movimiento creciente, es decir cuando $q_f > q_0$.

$w_m < w_0, w_f < 0$ Para un movimiento decreciente, es decir cuando $q_f < q_0$.

Si la velocidad inicial w_0 y final w_f son iguales a la velocidad media w_m , se obtiene un movimiento lineal en función del tiempo. Reemplazando w_0 y w_f por w_m en la ecuación (3.20) se obtiene una expresión de la velocidad igual a la ecuación (3.9).

Si la velocidad inicial w_0 y final w_f son iguales a cero se obtiene un movimiento angular en función del tiempo igual a:

$$q(t) = q_0 + (-2\tau + 3)\tau^2(q_f - q_0) \quad (3.22)$$

Evaluando en la expresión (3.21) con los valores de $\tau=0$ y $\tau=1$ se tiene el valor de la aceleración inicial α_0 y final α_f respectivamente como:

$$\alpha_0 = \alpha(0) = 6\frac{w_m}{T} - 2\frac{(2w_0 + w_f)}{T} \quad (3.23.a)$$

$$\alpha_f = \alpha(1) = -6\frac{w_m}{T} + 2\frac{(w_0 + 2w_f)}{T} \quad (3.23.b)$$

La velocidad máxima (si el movimiento es creciente) o mínima (si el movimiento es decreciente), se obtiene igualando a cero la expresión de la aceleración (3.21), obteniéndose el valor de τ_m igual a:

$$\tau_m = \frac{2w_0 + w_f - 3w_m}{3w_0 + 3w_f - 6w_m} \quad (3.24)$$

Reemplazando el valor de τ_m en la expresión de la velocidad (3.20) se obtiene la velocidad máxima/mínima como:

$$w_{max/min} = w_0 + \frac{(3w_m - 2w_0 - w_f)^2}{3(2w_m - w_0 - w_f)} \quad (3.25)$$

En forma gráfica la trayectoria para cada articulación con su respectiva velocidad y aceleración se aprecia en la Fig. (3.3).

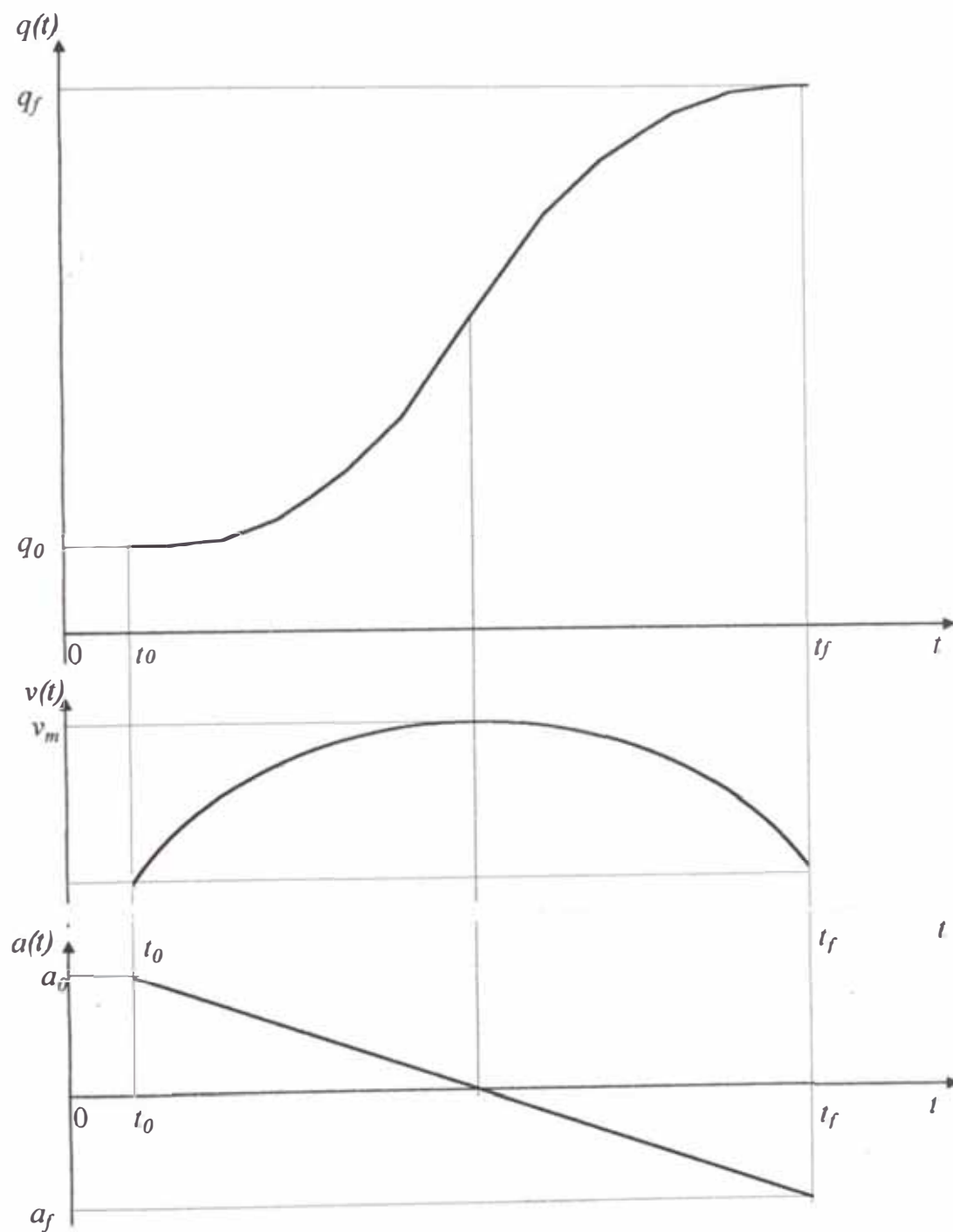


Fig. 3.3 Interpolación cúbica: posición, velocidad y aceleración.

La gran ventaja de utilizar este polinomio, como interpolación del movimiento de las articulaciones de un robot, es que la velocidad de la trayectoria aumenta desde que se inicia el movimiento con la velocidad inicial especificada (normalmente nula) hasta obtener la máxima magnitud alrededor del punto medio de la trayectoria y luego comienza a disminuir dicha velocidad hasta finalizar el movimiento con la velocidad final especificada (normalmente nula).

Las limitaciones de este polinomio estriba en la presencia de discontinuidades de la aceleración al inicio y al final de la trayectoria, lo cual da lugar a sacudidas en estos puntos sobre todo cuando se desea realizar conexiones de movimiento entre segmentos de trayectoria a grandes velocidades.

3.5.3 Interpolación polinómica de grado superior

Al usar interpolaciones lineales o cúbicas se presentan dificultades y limitaciones que impiden obtener una trayectoria totalmente controlada, tanto en posición, como en velocidad y aceleración.

En la planificación de una trayectoria de articulación interpolada para un robot hay que tener las siguientes consideraciones:

1. Cuando se toma un objeto, el movimiento de la mano debe dirigirse hacia fuera del objeto en otro caso, la mano puede golpear la superficie soporte del objeto.
2. Si especificamos una posición de partida (punto de despegue) a lo largo del vector normal a la superficie desde la posición inicial, y si queremos que la mano pase a través de esta posición, entonces tenemos un movimiento de partida admisible. Si especificamos más aun el tiempo necesario para alcanzar esta posición, podríamos controlar la velocidad con la cual se despega el objeto.

3. El mismo conjunto de requisitos de despegue para el movimiento del brazo, es también válido cuando nos acercamos a la posición final, de manera que se pueda obtener y controlar la dirección de aproximación correcta.
4. De lo anterior, tenemos cuatro posiciones para cada movimiento del brazo: inicial, despegue, asentamiento, y final. Gráficamente esto se muestra en la Fig. (3.4).
5. Ligaduras de posición:
 - a) *Posición inicial*: serán la velocidad y aceleración (normalmente nulas).
 - b) *Posición de despegue*: movimiento continuo a puntos intermedios.
 - c) *Posición de asentamiento*: lo mismo que para la posición de despegue.
 - d) *Posición final*: serán la velocidad y aceleración (normalmente nulas).
6. Además de estas ligaduras, los extremos de todas las trayectorias de la articulación deben estar dentro de sus límites físicos y geométricos.
7. Consideraciones temporales:
 - a) *Segmentos de trayectoria inicial y final*: el tiempo se basa sobre la velocidad de aproximación de la mano a y desde la superficie y es un valor fijo, que depende de las características de los actuadores de la articulación.
 - b) *Puntos intermedios o segmentos a mitad de trayectoria*: el tiempo se basa en la velocidad y aceleración máxima de las articulaciones, y se utiliza para normalizar el máximo de estos tiempos (el máximo tiempo de la articulación más lenta).

Las ligaduras para la planificación de trayectorias de articulación interpoladas son las siguientes:

Posición inicial:

1. Posición (dada).
2. Velocidad (dada, normalmente nula).

3. Aceleración (dada, normalmente nula).

Posición intermedia:

1. Posición de despegue (dada).
2. Posición de despegue (continua con el segmento de trayectoria previo).
3. Velocidad (continua con el segmento de trayectoria previo).
4. Aceleración (continua con el segmento de trayectoria previo).
5. Posición de asentamiento (dada).
6. Posición de asentamiento (continua con el siguiente segmento de trayectoria).
7. Velocidad (continua con el siguiente segmento de trayectoria).
8. Aceleración (continua con el siguiente segmento de trayectoria).

Posición final:

1. Posición (dada).
2. Velocidad (dada, normalmente nula).
3. Aceleración (dada, normalmente nula).

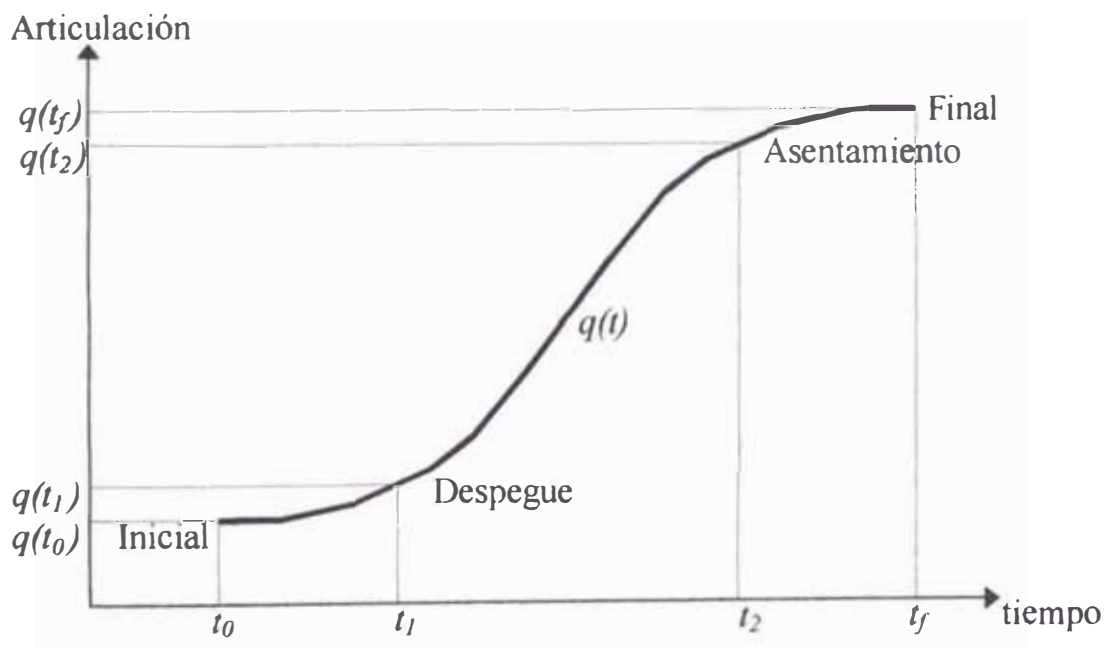


Fig. 3.4 Interpolación de orden superior.

Basándose en las ligaduras anteriores, estamos interesados en seleccionar una clase de funciones polinomiales de grado n o menor, tal que la posición, velocidad y aceleración de la articulación requerida en estos puntos nudos (inicial, despegue, asentamiento y final), se satisfagan, y la posición, velocidad y aceleración de la articulación sean continuas en el intervalo: $[t_0, t_f]$. Un método es especificar un polinomio $p(t)$ de séptimo grado para cada articulación:

$$p(t) = a_7t^7 + a_6t^6 + a_5t^5 + a_4t^4 + a_3t^3 + a_2t^2 + a_1t + a_0 \quad (3.26)$$

Los coeficientes incógnitas a_i se pueden determinar a partir de las posiciones conocidas y de las condiciones de continuidad. Sin embargo, la utilización de un polinomio de grado elevado para interpolar los puntos nudos dados, puede no ser satisfactorio. Es difícil encontrar sus puntos extremos y tienden a tener movimientos extraños. Un método alternativo es dividir la trayectoria completa de la articulación en algunos segmentos, de manera que se puedan utilizar diferentes polinomios de interpolación para cada uno de ellos. Existen diferentes maneras de dividir la trayectoria de la articulación, y cada método posee propiedades diferentes. Entre los más comunes son los siguientes:

- **trayectoria 4-3-4.** Cada articulación se planifica en base de tres segmentos: el primer segmento es un polinomio de cuarto orden; el segundo segmento es un polinomio de tercer orden y el tercer segmento es un polinomio de cuarto orden.
- **Trayectoria 3-5-3.** Utiliza polinomios de tercer, quinto y tercer grado para el primer, segundo y tercer segmento respectivamente.
- **Trayectoria con 5 cúbicas.** Se utilizan polinomios "spline" cúbicos de tercer grado para los cinco segmentos de trayectoria.

Aquí se desarrollará la planificación de una trayectoria de articulación 4-3-4 correspondiente a cada articulación, la cual está seccionada en tres segmentos:

- El primer segmento es un polinomio de cuarto orden que especifica la trayectoria desde la posición inicial hasta la de despegue.
- El segundo segmento de trayectoria (segmento medio de trayectoria) es un polinomio de tercer grado que especifica la trayectoria desde la posición de despegue hasta la posición de asentamiento.
- El tercer segmento de trayectoria es un polinomio de cuarto orden que especifica la trayectoria desde la posición de asentamiento hasta la posición final.

La trayectoria consiste en la secuencia de polinomios, $p(t)$, que juntos forman la trayectoria para cada articulación del robot. Las ecuaciones de los polinomios para cada variable de articulación en función del tiempo τ normalizado son:

$$p_1(\tau) = a_{14}\tau^4 + a_{13}\tau^3 + a_{12}\tau^2 + a_{11}\tau + a_{10} \quad (3.27.a)$$

$$p_2(\tau) = a_{23}\tau^3 + a_{22}\tau^2 + a_{21}\tau + a_{20} \quad (3.27.b)$$

$$p_3(\tau) = a_{34}\tau^4 + a_{33}\tau^3 + a_{32}\tau^2 + a_{31}\tau + a_{30} \quad (3.27.c)$$

donde:

$p_1(t)$ es el 1.^{er} segmento de la trayectoria.

$p_2(t)$ es el 2.^{do} segmento de la trayectoria.

$p_3(t)$ es el 3.^{er} segmento de la trayectoria.

El número de polinomios para una trayectoria 3-4-3 de un robot con n articulaciones tendrá n trayectorias de articulación, o sea $3n$ segmentos de trayectoria y $14n$ coeficientes de polinomios a evaluar cuya solución para los coeficientes a_{ij} de estos polinomios se encuentra desarrollado en el Apéndice B.

La gran ventaja de usar este tipo de trayectoria radica en la continuidad, tanto en

la velocidad, como de la aceleración en los puntos nudos (despegue y asentamiento). Esto hace que no se presenten "sacudidas" o sobresaltos; es decir, la velocidad del cambio de aceleración es también continuo en los puntos nudos de la trayectoria.

Otra de las ventajas es que se puede asignar velocidades y aceleraciones, ya sea en la posición inicial y final para conectar movimientos con otros segmentos de trayectoria.

La desventaja de usar este tipo de interpolación es que el segundo polinomio de tercer orden presenta puntos de inflexión, lo cual hace que la aceleración aumente y disminuya, haciendo que la velocidad tenga sentido diferente al movimiento de la trayectoria, provocando avances y retrocesos en el movimiento, lo cual no debe ocurrir en la trayectoria de un robot. También provoca en las articulaciones valores superiores a la posición final q_f y/o inferiores a posición inicial q_0 , lo que puede alcanzar valores q que están fuera del rango de las articulaciones y del ámbito de trabajo del espacio de articulación.

Otra de las desventajas se presenta en la evaluación de los coeficientes de los polinomios correspondientes a cada segmento, por el hecho de que la matriz C expresada mediante la ecuación (B.27) del Apéndice B, no es diagonalmente dominante, lo cual origina valores de a_{ij} , grandes e inexactos.

En conclusión cuando el robot va a operar a bajas velocidades, se suele utilizar una trayectoria con interpolación lineal, para velocidades medias y controladas se puede utilizar una trayectoria con interpolación cubica y para velocidades muy grandes y con aceleraciones a controlar se utilizarán trayectorias con interpolaciones de grado superior.

3.6 Generación de trayectorias cartesianas

Para la planificación en el espacio cartesiano se define la historia temporal de la posición del efector final del manipulador, su velocidad y su aceleración, y se deducen las correspondientes posiciones, velocidades y aceleraciones de las articulaciones a partir de la información del efector final.

Generalmente, la planificación de la trayectoria en el espacio cartesiano se puede conseguir en dos pasos coherentes:

1. Generación de un conjunto de puntos nudos o puntos de interpolación en coordenadas cartesianas, siguiendo algunas reglas a lo largo del camino cartesiano.
2. Especificación de una clase de funciones, para enlazar estos puntos nudos de acuerdo con algún criterio.

En la sección anterior, se describía funciones polinomiales de diferente orden para generar puntos de consignas de la trayectoria de la articulación interpolada en el control del manipulador. Aunque las coordenadas de articulación del manipulador especifican completamente la posición y orientación de su efector final, no son apropiadas para especificar una determinada tarea, porque la mayoría de las coordenadas de articulación del manipulador no son ortogonales y no separan la posición de la orientación. Para los robots más sofisticados, se desarrollan lenguajes de programación que controlan al manipulador, para cumplir una tarea específica en el sistema de coordenadas cartesianas. En tales sistemas, una tarea se suele especificar como secuencias de puntos nudos cartesianos, a través de los cuales debe pasar el efector final del manipulador. Así, al describir los movimientos del manipulador en una tarea, interesa el formalismo de especificar las posiciones

finales, por las cuales tiene que moverse el efector final del robot, así como la curva en el espacio o camino que debe recorrer.

Aquí se describe el diseño de las trayectorias del manipulador constituidas por segmentos de línea recta para el movimiento del efector final. La velocidad y aceleración del efector final entre estos segmentos se controla, convirtiéndolas en coordenadas de articulación y se suavizan mediante una rutina de interpolación cuadrática.

Los robots industriales, se suelen controlar en el espacio de las variables de articulación, mientras que los objetos que se manipulan se suelen expresar en el sistema de coordenadas de referencia.

El movimiento en línea recta de un robot, representa la distancia más corta entre los puntos en el espacio de trabajo. Hay una variedad de aplicaciones en la cual la herramienta, es requerida para seguir o al menos aproximarse estrechamente a una trayectoria recta. Por ejemplo, el rastreo de una cinta transportadora, seguimiento de la costura en línea recta durante la soldadura por arco, la inserción de un tornillo en un agujero, etc. En muchas de estas aplicaciones, el camino en línea recta es realizado a una velocidad constante. Estas son las instancias del movimiento uniforme en línea recta, donde las fuerzas inerciales en la carga del efector final deben ser minimizadas.

Para planificar y controlar una trayectoria en línea recta, en el espacio de coordenadas de referencia que pueden ser cartesianas, se usan las ecuaciones de la cinemática inversa que sirven para determinar la trayectoria en el espacio de articulación, la cual genera la trayectoria en el espacio cartesiano mediante la cinemática directa.

Existen varios métodos para generar movimientos en línea recta respecto del sistema de coordenadas de referencia. Los métodos que desarrollaremos aquí son: interpolación lineal del vector de variables cartesianas, interpolación lineal mediante transformación homogénea e interpolación lineal de rotación y traslación.

3.6.1 Interpolación lineal del vector de variables cartesianas

Una clase importante de trayectoria en línea recta que pueden seguir algunos manipuladores corresponde al caso del movimiento del extremo \mathbf{p} del efector final a lo largo de una línea recta definida por un vector de aproximación constante (\mathbf{a}). Este tipo de movimiento incluye por ejemplo la inserción y extracción en la operación de armar y desarmar máquinas dentro del espacio de trabajo. Para muchos robots, solamente el control del movimiento es factible, porque la velocidad de las articulaciones individuales no se pueden variar independientemente, o se pueden aproximar a una trayectoria en línea recta, registrando una secuencia apropiada de puntos conocidos en el espacio de articulación.

El movimiento del manipulador en las coordenadas cartesianas, entre la localización inicial \mathbf{F}_0 y final \mathbf{F}_f es visto y realizado sobre un segmento de línea recta espacial, cuya velocidad puede ser constante o adoptar alguna forma parabólica. Si \mathbf{x}_0 y \mathbf{x}_f denotan un punto inicial y final respectivamente, en el espacio cartesiano, o la configuración del efector final en \mathbf{R}^6 o sea se tiene:

$$\mathbf{x}_0 = \begin{bmatrix} p_{x0} & p_{y0} & p_{z0} & \varphi_0 & \theta_0 & \psi_0 \end{bmatrix}^T \quad (3.28)$$

$$\mathbf{x}_f = \begin{bmatrix} p_{xf} & p_{yf} & p_{zf} & \varphi_f & \theta_f & \psi_f \end{bmatrix}^T \quad (3.29)$$

Si el movimiento a ser recorrido fuera realizado en T segundos, entonces la trayectoria general en línea recta para la herramienta en función del tiempo se puede representar como:

$$x(t) = x_0 + \lambda(t)(x_f - x_0) \quad (3.30)$$

La expresión de la velocidad $v(t)$ se obtiene derivando la ecuación (3.30):

$$v(t) = \frac{dx(t)}{dt} = \frac{d\lambda(t)}{dt}(x_f - x_0) = \frac{(x_f - x_0)}{T} \quad (3.31)$$

La expresión $\lambda(\tau)$ está dada por:

$$\lambda(t) = \frac{t - t_0}{t_f - t_0} = \frac{t - t_0}{T} \quad (3.32)$$

donde:

λ : tiempo normalizado, $\lambda \in [0, 1]$.

t : tiempo real en segundos, $t \in [t_0, t_f]$

t_0 : tiempo real al inicio de la trayectoria.

t_f : tiempo real al final de la trayectoria.

$T = t_f - t_0$: tiempo real necesitado para recorrer toda la trayectoria.

x_0 : localización inicial del efector final.

x_f : localización final del efector final.

A cada punto de $x(t)$ de la ecuación (3.30) le corresponde un determinado valor de $q(t)$; por lo que se aplicará la cinemática inversa para evaluar las magnitudes de las variables articulares, determinando así una trayectoria en el espacio articular dependiente de la trayectoria cartesiana. También se puede usar un polinomio $x(t)$ de tercer grado como en la ecuación (3.10).

Desde que las ecuaciones de la cinemática inversa han de resolverse para cada punto conocido de $x(t)$, es deseable minimizar un número conocido de puntos y distribuirlos a lo largo de la trayectoria programada de una manera óptima. La desviación de error de la trayectoria puede ser evaluada en cada punto y si excede una tolerancia prescrita, entonces se debe corregir de forma inmediata para obtener una trayectoria prácticamente sin desviaciones.

3.6.2 Interpolación lineal mediante transformación homogénea

En un sistema robótico programable, el movimiento deseado se puede especificar como una secuencia de puntos nudos en el sistema de coordenadas cartesianas, cada uno de los cuales se puede describir en términos de las transformaciones homogéneas, que relacionan el sistema de coordenadas de la mano del manipulador con el sistema de coordenadas del espacio de trabajo. Las correspondientes coordenadas de articulación en estos puntos nudos cartesianos, se pueden calcular a partir de la rutina de la solución del problema cinemático inverso, y se puede utilizar un polinomio cuadrático para suavizar, con fines de control, los puntos nudos de dos articulaciones consecutivas en coordenadas de articulación. Así, se controla el efector final del manipulador para moverse a lo largo de una línea recta conectada mediante estos puntos nudos. Esta técnica tiene la ventaja de permitirnos controlar el efector final del robot para seguir a objetos móviles. Aunque las posiciones finales se describen mediante sus transformadas, no especifican cómo se mueve la mano o efector final del robot de un punto transformado a otro. Para lograr esto, se utiliza una traslación en línea recta y dos rotaciones para conseguir el movimiento entre dos puntos cartesianos consecutivos.

En general, las posiciones finales del robot y del objetivo se pueden expresar en la ecuación matricial fundamental siguiente:

$${}^0\mathbf{T}^6 {}^6\mathbf{T}^{\text{herr}} \mathbf{D}^{\text{obj}}(t) = {}^0\mathbf{C}^{\text{base}}(t) {}_{\text{base}}\mathbf{P}^{\text{obj}} \quad (3.33)$$

donde:

${}^0\mathbf{T}^6$ = Matriz de transformación homogénea 4×4 que describe la posición y orientación del efector final del manipulador con respecto al referencial de la base.

${}^6\mathbf{T}^{\text{herr}}$ = Matriz de transformación homogénea 4×4 que describe la posición y orientación del punto final de la herramienta con respecto al referencial del efector final. Describe el punto final de la herramienta cuyo movimiento se desea controlar.

${}^0\mathbf{C}^{\text{base}}(t)$ = Matriz de transformación homogénea 4×4 en función del tiempo que describe el referencial de trabajo del objeto con respecto al referencial de la base.

${}_{\text{base}}\mathbf{P}^{\text{obj}}$ = Matriz de transformación homogénea 4×4 que describe la posición y orientación deseada del objeto en movimiento, con respecto al referencial de trabajo.

${}_{\text{herr}}\mathbf{D}^{\text{obj}}(t)$ = Matriz de transformación homogénea 4×4 en función del tiempo denominada matriz de "impulsión" que describe la trayectoria a seguir en función del tiempo, con el fin que la herramienta pueda alcanzar al objeto. Esta matriz representa la localización del objeto con respecto a la localización del punto final de la herramienta.

En la ecuación (3.33), las matrices del lado izquierdo describen la posición y orientación de la pinza del manipulador y la trayectoria a seguir, mientras que las matrices del lado derecho, describen la posición y orientación de la característica del objeto donde nos gustaría que la herramienta del manipulador lo cogiese.

Si se combina ${}^6\mathbf{T}^{\text{herr}}$ con ${}^0\mathbf{T}^6$ para formar la matriz del brazo, entonces ${}^6\mathbf{T}^{\text{herr}}$ es una

matriz identidad 4×4 y se puede omitir. Si el referencial de trabajo es el mismo que el referencial de la base del manipulador, entonces ${}_0\mathbf{C}^{\text{hase}}(t)$ es una matriz identidad 4×4 en todo instante del tiempo; entonces la expresión (3.33) se puede re-escribir como:

$${}_0\mathbf{T}^{\text{herr}} \mathbf{D}_{\text{herr}}^{\text{obj}}(t) = {}_0\mathbf{P}^{\text{obj}} \quad (3.34)$$

Así se puede resolver para ${}_0\mathbf{T}^{\text{herr}}$, matriz que describe la configuración del manipulador para coger el objeto de una forma correcta y deseada:

$${}_0\mathbf{T}^{\text{herr}} = {}_0\mathbf{P}^{\text{obj}} \left[\mathbf{D}_{\text{herr}}^{\text{obj}}(t) \right]^{-1} \quad (3.35)$$

Si ${}_0\mathbf{T}^{\text{herr}}$ se evalúe a una velocidad suficientemente alta y al convertirse en los correspondientes ángulos de articulación, el manipulador se podría controlar para seguir la trayectoria planificada en la configuración cartesiana.

Si se dan dos puntos fijos p_i y p_{i+1} (en un determinado instante) y las velocidades lineales y angulares, entonces se puede determinar las distancias y el tiempo T para ir de la localización i a la localización $i+1$. Como las herramientas y los referenciales de objetos móviles se especifican con respecto al referencial base, el movimiento se puede hacerlo mejor especificándolo con respecto al referencial destino. Esto tiene la ventaja de que la herramienta parece que está en reposo desde el sistema de coordenadas del objeto móvil.

Nuestro análisis se realizará con respecto a la base considerando que el objeto está fijo. El movimiento desde la localización F_i hasta la localización F_{i+1} se expresa en términos del tiempo t por medio de una transformación de "impulsión" $\mathbf{D}(\lambda)$, λ es una forma de tiempo normalizado dependiendo del tiempo t , dada por la ecuación (3.32). Volviendo a escribir la ecuación (3.34) se obtiene:

$${}_0\mathbf{F}_{i+1}^{\text{obj}}(\lambda) = {}_0\mathbf{F}_i^{\text{herr}} \mathbf{D}^{\text{obj}}(\lambda) \quad (3.36)$$

En la posición inicial λ es cero. Por tanto, $\mathbf{D}(0)$ será una matriz de identidad 4×4 y en la posición final λ vale 1 por consiguiente se obtiene:

$$\mathbf{F}_{i+1} = \mathbf{F}_i \mathbf{D}(1) \quad (3.37)$$

Despejando $\mathbf{D}(1)$ obtenemos:

$$\mathbf{D}(1) = (\mathbf{F}_i)^{-1} \mathbf{F}_{i+1} \quad (3.38)$$

Expresando las localizaciones \mathbf{F}_i y \mathbf{F}_{i+1} mediante sus respectivas matrices de transformación homogénea se obtiene:

$$\mathbf{F}_i \triangleq \mathbf{A} = \begin{bmatrix} \mathbf{n}_A & \mathbf{s}_A & \mathbf{a}_A & \mathbf{p}_A \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.39)$$

$$\mathbf{F}_{i+1} \triangleq \mathbf{B} = \begin{bmatrix} \mathbf{n}_B & \mathbf{s}_B & \mathbf{a}_B & \mathbf{p}_B \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.40)$$

Utilizando la ecuación (A.14) del Anexo A para invertir \mathbf{F}_i y multiplicar por \mathbf{F}_{i+1} en la ecuación (3.38) se obtiene:

$$\mathbf{D}(1) = \begin{bmatrix} \mathbf{R}_A^T & -\mathbf{R}_A^T \mathbf{p}_A \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_B & \mathbf{p}_B \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_A^T \mathbf{R}_B & \mathbf{R}_A^T (\mathbf{p}_B - \mathbf{p}_A) \\ 0 & 1 \end{bmatrix} \quad (3.41)$$

Desarrollando la ecuación (3.41) se obtiene la matriz de impulsión como:

$$\mathbf{D}(1) = \begin{bmatrix} \mathbf{n}_A \cdot \mathbf{n}_B & \mathbf{n}_A \cdot \mathbf{s}_B & \mathbf{n}_A \cdot \mathbf{a}_B & \mathbf{n}_A \cdot (\mathbf{p}_B - \mathbf{p}_A) \\ \mathbf{s}_A \cdot \mathbf{n}_B & \mathbf{s}_A \cdot \mathbf{s}_B & \mathbf{s}_A \cdot \mathbf{a}_B & \mathbf{s}_A \cdot (\mathbf{p}_B - \mathbf{p}_A) \\ \mathbf{a}_A \cdot \mathbf{n}_B & \mathbf{a}_A \cdot \mathbf{s}_B & \mathbf{a}_A \cdot \mathbf{a}_B & \mathbf{a}_A \cdot (\mathbf{p}_B - \mathbf{p}_A) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.42)$$

Si la función impulsión consiste en un movimiento traslacional y dos movimientos rotacionales, entonces ambos serán directamente proporcionales a λ . Si λ varía linealmente con el tiempo, entonces el movimiento resultante representado por $\mathbf{D}(\lambda)$

corresponderá a una velocidad lineal y dos velocidades angulares constantes. Así la función de impulsión se puede representar como:

$$\mathbf{D}(\lambda) = \mathbf{L}(\lambda) \mathbf{R}_A(\lambda) \mathbf{R}_B(\lambda) \quad (3.43)$$

El movimiento traslacional se puede representar mediante una matriz de transformación homogénea $\mathbf{L}(\lambda)$ y el movimiento será a lo largo de la línea recta llevando al efector final del punto \mathbf{p}_i al punto \mathbf{p}_{i+1} .

$$\mathbf{L}(\lambda) = \begin{bmatrix} 1 & 0 & 0 & \lambda x \\ 0 & 1 & 0 & \lambda y \\ 0 & 0 & 1 & \lambda z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.44)$$

El primer movimiento rotacional se puede representar mediante una matriz de transformación homogénea $\mathbf{R}_A(\lambda)$ y sirve para rotar el vector de aproximación de \mathbf{F}_i hasta el vector de aproximación en \mathbf{F}_{i+1} , alineando de este modo el efector final a lo largo del vector de aproximación deseado como sigue:

$$\mathbf{R}_A(\lambda) = \begin{bmatrix} S^2\psi V(\lambda\theta) + C(\lambda\theta) & -S\psi C\psi V(\lambda\theta) & C\psi S(\lambda\theta) & 0 \\ -S\psi C\psi V(\lambda\theta) & C^2\psi V(\lambda\theta) + C(\lambda\theta) & S\psi S(\lambda\theta) & 0 \\ -C\psi S(\lambda\theta) & -S\psi S(\lambda\theta) & C(\lambda\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.45)$$

donde:

$$V(\lambda\theta) = \text{Verseno}(\lambda\theta) = 1 - \cos(\lambda\theta)$$

$$C(\lambda\theta) = \cos(\lambda\theta) \quad S(\lambda\theta) = \text{sen}(\lambda\theta)$$

$$C\psi = \cos(\psi) \quad S\psi = \text{sen}(\psi)$$

Matemáticamente, la matriz de rotación $\mathbf{R}_A(\lambda)$ indica una rotación de ángulo θ respecto del vector de orientación o deslizamiento (\mathbf{s}) del efector final de \mathbf{F}_i que ha

girado un ángulo ψ respecto del vector de aproximación (\mathbf{a}). O sea en la matriz de rotación $\mathbf{R}_A(\lambda)$, el ángulo que gira es θ y el vector unitario de orientación, alrededor del cual gira es $[-S\psi \ C\psi \ 0]^T$, que viene a ser una rotación $Rot(\mathbf{a}, \psi)$ con respecto al eje de aproximación. Esto es:

$$Rot(\mathbf{a}, \psi) = \begin{bmatrix} C\psi & -S\psi & 0 \\ S\psi & C\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.46)$$

El segundo movimiento rotacional, representado por $\mathbf{R}_B(\lambda)$, sirve para rotar el vector de orientación de \mathbf{F}_i en el vector de orientación en \mathbf{F}_{i+1} respecto del eje del vector de aproximación. Físicamente, $\mathbf{R}_B(\lambda)$ representa una rotación de ϕ con respecto del vector de aproximación del efector final en \mathbf{F}_{i+1} .

$$\mathbf{R}_B(\lambda) = \begin{bmatrix} C(\lambda\phi) & -S(\lambda\phi) & 0 & 0 \\ S(\lambda\phi) & C(\lambda\phi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.47)$$

donde:

$$C(\lambda\phi) = \cos(\lambda\phi) \quad S(\lambda\phi) = \text{sen}(\lambda\phi)$$

Multiplicando las matrices de las ecuaciones (3.44) (3.45) y (3.47) juntas tenemos:

$$\mathbf{D}(\lambda) = \begin{bmatrix} \mathbf{dn}(\lambda) & \mathbf{ds}(\lambda) & \mathbf{da}(\lambda) & \mathbf{dp}(\lambda) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.48)$$

donde:

$$\mathbf{dn}(\lambda) = \begin{bmatrix} C(\lambda\phi)[S^2\psi V(\lambda\theta) + C(\lambda\theta)] + S(\lambda\phi)[-S\psi C\psi V(\lambda\theta)] \\ C(\lambda\phi)[-S\psi C\psi V(\lambda\theta)] + S(\lambda\phi)[C^2\psi V(\lambda\theta) + C(\lambda\theta)] \\ C(\lambda\phi)[-C\psi S(\lambda\theta)] + S(\lambda\phi)[-S\psi S(\lambda\theta)] \end{bmatrix} \quad (3.49.a)$$

$$\mathbf{ds}(\lambda) = \begin{bmatrix} -S(\lambda\phi)[S^2\psi V(\lambda\theta) + C(\lambda\theta)] + C(\lambda\phi)[-S\psi C\psi V(\lambda\theta)] \\ -S(\lambda\phi)[-S\psi C\psi V(\lambda\theta)] + C(\lambda\phi)[C^2\psi V(\lambda\theta) + C(\lambda\theta)] \\ -S(\lambda\phi)[-C\psi S(\lambda\theta)] + C(\lambda\phi)[-S\psi S(\lambda\theta)] \end{bmatrix} \quad (3.49.b)$$

$$\mathbf{da}(\lambda) = \begin{bmatrix} C\psi S(\lambda\theta) \\ S\psi S(\lambda\theta) \\ C(\lambda\theta) \end{bmatrix} \quad (3.49.c)$$

$$\mathbf{dp}(\lambda) = \begin{bmatrix} \lambda x \\ \lambda y \\ \lambda z \end{bmatrix} \quad (3.49.d)$$

Utilizando la técnica de la transformada inversa sobre la ecuación (3.43) se puede resolver para x , y , y z postmultiplicando la ecuación (3.43) por $\mathbf{R}_B^{-1}(\lambda) \mathbf{R}_A^{-1}(\lambda)$ e igualando los elementos del vector posición:

$$\begin{aligned} x &= \mathbf{n}_A \cdot (\mathbf{p}_B - \mathbf{p}_A) \\ y &= \mathbf{s}_A \cdot (\mathbf{p}_B - \mathbf{p}_A) \\ z &= \mathbf{a}_A \cdot (\mathbf{p}_B - \mathbf{p}_A) \end{aligned} \quad (3.50)$$

Al postmultiplicar ambos lados de la ecuación (3.43) por $\mathbf{R}_B^{-1}(\lambda)$ y a continuación premultiplicar por $\mathbf{L}^{-1}(\lambda)$, se obtiene θ y ψ igualando los elementos de la tercera columna con los elementos de la tercera columna de la ecuación (3.43) como sigue:

$$\psi = \text{tg}^{-1} \left(\frac{\mathbf{s}_A \cdot \mathbf{a}_B}{\mathbf{n}_A \cdot \mathbf{a}_B} \right); \quad -\pi \leq \psi \leq \pi \quad (3.51)$$

$$\theta = \text{tg}^{-1} \left(\frac{\left[(\mathbf{n}_A \cdot \mathbf{a}_B)^2 + (\mathbf{s}_A \cdot \mathbf{a}_B)^2 \right]^{\frac{1}{2}}}{\mathbf{a}_A \cdot \mathbf{a}_B} \right); \quad 0 \leq \theta \leq \pi \quad (3.52)$$

Para determinar ϕ premultiplicamos ambos lados de la ecuación (3.43) por $\mathbf{L}^{-1}(\lambda)$ y a continuación por $\mathbf{R}_A^{-1}(\lambda)$ e igualar los elementos respectivos para obtener :

$$\phi = \operatorname{tg}^{-1} \left\{ \frac{K_1(\mathbf{n}_A \cdot \mathbf{n}_B) + K_2(\mathbf{s}_A \cdot \mathbf{n}_B) + K_3(\mathbf{a}_A \cdot \mathbf{n}_B)}{K_1(\mathbf{n}_A \cdot \mathbf{s}_B) + K_2(\mathbf{s}_A \cdot \mathbf{s}_B) + K_3(\mathbf{a}_A \cdot \mathbf{s}_B)} \right\}, \quad -\pi \leq \phi < \pi \quad (3.53)$$

donde:

$$\begin{aligned} K_1 &= -S\psi C\psi V(\lambda\theta) \\ K_2 &= C^2\psi V(\lambda\theta) + C(\lambda\theta) \\ K_3 &= -S\psi S(\lambda\theta) \end{aligned} \quad (3.54)$$

3.6.3 Interpolación lineal de rotación y traslación

Si se requiere mover el efector final de un manipulador a lo largo de una línea recta entre dos puntos nudos especificados por \mathbf{F}_1 y \mathbf{F}_2 en el tiempo T , donde cada localización se representa mediante una matriz de transformación homogénea:

$$\mathbf{F}_i = \begin{bmatrix} \mathbf{R}_i & \mathbf{p}_i \\ 0 & 1 \end{bmatrix} \quad (3.55)$$

Entonces el movimiento a lo largo del camino, consiste en una traslación del referencial del efector final del robot desde \mathbf{p}_1 hasta \mathbf{p}_2 junto con una rotación desde \mathbf{R}_1 hasta \mathbf{R}_2 . Sea $\lambda(t)$ la variable de la ecuación (3.32) que representa la fracción del movimiento para ser recorrida desde el tiempo t_0 hasta el tiempo t para un determinado movimiento. La posición $\mathbf{p}(t)$ y orientación $\mathbf{R}(t)$ del efector final en el tiempo t se expresan respectivamente por:

$$\mathbf{p}(t) = \mathbf{p}_1 + \lambda(t)(\mathbf{p}_2 - \mathbf{p}_1) \quad (3.56)$$

$$\mathbf{R}(t) = \mathbf{R}_1 \operatorname{Rot}(\mathbf{u}, \lambda(t)\theta) \quad (3.57)$$

Realizando adecuadamente operaciones elementales en las ecuaciones (3.60) y considerando que \mathbf{u} es un vector unitario se llega a determinar:

$$S\theta = \frac{\sqrt{(\mathbf{a}_A \mathbf{s}_B - \mathbf{s}_A \mathbf{a}_B)^2 + (\mathbf{n}_A \mathbf{a}_B - \mathbf{a}_A \mathbf{n}_B)^2 + (\mathbf{s}_A \mathbf{n}_B - \mathbf{n}_A \mathbf{s}_B)^2}}{2} \quad (3.61)$$

$$C\theta = \frac{\mathbf{n}_A \mathbf{n}_B + \mathbf{s}_A \mathbf{s}_B + \mathbf{a}_A \mathbf{a}_B - 1}{2} \quad (3.62)$$

El ángulo θ y el vector \mathbf{u} se expresan respectivamente por:

$$\theta = \arctan\left(\frac{S\theta}{C\theta}\right) \quad (3.63)$$

$$\mathbf{u} = \left[\frac{\mathbf{a}_A \mathbf{s}_B - \mathbf{s}_A \mathbf{a}_B}{2 S\theta} \quad \frac{\mathbf{n}_A \mathbf{a}_B - \mathbf{a}_A \mathbf{n}_B}{2 S\theta} \quad \frac{\mathbf{s}_A \mathbf{n}_B - \mathbf{n}_A \mathbf{s}_B}{2 S\theta} \right]^T \quad (3.64)$$

Es conveniente indicar que $\mathbf{p}_2 - \mathbf{p}_1$ en la ecuación (3.56) y \mathbf{u} y θ en la ecuación (3.57) necesitan evaluarse solamente una vez por segmento si el punto \mathbf{F}_2 está fijo. Por el contrario, si el segmento de destino está cambiando, \mathbf{F}_2 estará también modificándose. En este caso $\mathbf{p}_2 - \mathbf{p}_1$, \mathbf{u} y θ se deberán evaluar cada vez que cambie \mathbf{F}_2 .

3.7 Consideraciones finales

Tal como se ha mostrado a lo largo de este capítulo, el control del movimiento de la estructura articulada de un robot es un problema complejo que exige el planeamiento de estrategias elaboradas para lograr un funcionamiento rápido y preciso del robot.

Las estrategias expuestas en los apartados precedentes tienen carácter general, lo que incrementa su complejidad y dificultad de utilización en tiempo real. A la hora de aplicar cualquiera de estas técnicas a un determinado robot, pueden ser múltiples las simplificaciones y variantes que será posible introducir con el fin de conseguir un

donde $Rot(\mathbf{u}, \theta)$ es una rotación de un ángulo θ respecto de un eje o vector unitario \mathbf{u} para orientar al efector final desde \mathbf{R}_1 hasta \mathbf{R}_2 y matricialmente se expresa por:

$$Rot(\mathbf{u}, \theta) = \mathbf{R}_1^{-1} \mathbf{R}_2$$

Utilizando la ecuación (A.13) del Anexo A, que representa una rotación de un ángulo θ respecto de un vector arbitrario \mathbf{u} , y utilizando la matriz de rotación \mathbf{R} de la expresión (3.42) de la función de impulsión $\mathbf{D}(\lambda)$, respectivamente se obtiene:

$$Rot(\mathbf{u}, \theta) = \begin{bmatrix} u_x^2 V \theta + C \theta & u_x u_y V \theta - u_z S \theta & u_x u_z V \theta + u_y S \theta \\ u_x u_y V \theta + u_z S \theta & u_y^2 V \theta + C \theta & u_y u_z V \theta - u_x S \theta \\ u_x u_z V \theta - u_y S \theta & u_y u_z V \theta + u_x S \theta & u_z^2 V \theta + C \theta \end{bmatrix} \quad (3.58)$$

$$\mathbf{R}_1^{-1} \mathbf{R}_2 = \begin{bmatrix} \mathbf{n}_A \cdot \mathbf{n}_B & \mathbf{n}_A \cdot \mathbf{s}_B & \mathbf{n}_A \cdot \mathbf{a}_B \\ \mathbf{s}_A \cdot \mathbf{n}_B & \mathbf{s}_A \cdot \mathbf{s}_B & \mathbf{s}_A \cdot \mathbf{a}_B \\ \mathbf{a}_A \cdot \mathbf{n}_B & \mathbf{a}_A \cdot \mathbf{s}_B & \mathbf{a}_A \cdot \mathbf{a}_B \end{bmatrix} \quad (3.59)$$

Igualando término a término las componentes de los extremos de la derecha de las expresiones (3.58) y (3.59) se obtiene:

$$\begin{aligned} u_x^2 V \theta + C \theta &= \mathbf{n}_A \cdot \mathbf{n}_B \\ u_x u_y V \theta + u_z S \theta &= \mathbf{s}_A \cdot \mathbf{n}_B \\ u_x u_z V \theta - u_y S \theta &= \mathbf{a}_A \cdot \mathbf{n}_B \\ u_x u_y V \theta - u_z S \theta &= \mathbf{n}_A \cdot \mathbf{s}_B \\ u_y^2 V \theta + C \theta &= \mathbf{s}_A \cdot \mathbf{s}_B \\ u_y u_z V \theta + u_x S \theta &= \mathbf{a}_A \cdot \mathbf{s}_B \\ u_x u_z V \theta + u_y S \theta &= \mathbf{n}_A \cdot \mathbf{a}_B \\ u_y u_z V \theta - u_x S \theta &= \mathbf{s}_A \cdot \mathbf{a}_B \\ u_z^2 V \theta + C \theta &= \mathbf{a}_A \cdot \mathbf{a}_B \end{aligned} \quad (3.60)$$

compromiso aceptable entre eficacia y complejidad. La habilidad, por tanto del diseñador del sistema de control de un robot estará precisamente en realizar un diseño “a medida” que saque el máximo partido de las características particulares de dicho robot, modificando y adaptando adecuadamente los métodos generales indicados

Además del control de la posición y orientación en el tiempo del elemento terminal cabe mencionar, por su importancia en las aplicaciones de ensamblaje, el control del torque (o fuerza) y el control de acomodación.

Las trayectorias que se han desarrollado para el *ROB-DEX* son tanto con respecto a la articulación como con respecto al sistema de referencia. Para controlar la trayectoria del ROB-DEX a nivel de articulación se ha utilizado interpolaciones polinómicas lineal y cúbica en forma simultánea a las cuatro articulaciones.

Con respecto al sistema de referencia el *ROB-DEX* realiza trayectorias en línea recta usando interpolaciones de variables cartesianas de forma lineal y cúbica.

CAPITULO IV

DINAMICA DEL ROBOT

4.1 Introducción

La dinámica del robot trata con las formulaciones físicas o matemáticas del movimiento que realiza el manipulador, considerando las fuerzas que lo originan. Las ecuaciones físicas del sistema describen su conducta dinámica en el tiempo y sirven para la simulación del movimiento del robot en el computador, para la evaluación del diseño y estructura del robot y para diseñar los servomecanismos que impulsan a las articulaciones.

Con el conocimiento de la cinemática (directa o inversa) del robot, es posible controlar los movimientos de un robot, utilizando estrategias de control basados únicamente en la cinemática, en general cuando las velocidades de desplazamiento no son muy grandes (no mayores al 20% de la frecuencia propia del mecanismo) se dice que el robot está mecánicamente sobredimensionado. Al aumentar la velocidad de trabajo (arriba del 50% de la frecuencia propia del robot) aparecen fenómenos dinámicos, complejos no contemplados en los modelos cinemáticos. Tales fenómenos pueden anularse, obteniéndose un comportamiento óptimo, utilizando estrategias de control sofisticadas. Sin embargo, para tener un buen control del robot moviéndose a grandes velocidades, es necesario conocer la dinámica completa y en forma exacta de todo el mecanismo del manipulador.

4.2 Modelamiento dinámico

El modelamiento dinámico o matemático de un robot se puede obtener a partir de leyes físicas, tales como las de la mecánica *newtoniana* y *lagrangeana*. Esto conduce al desarrollo de las ecuaciones que describen el movimiento dinámico de las articulaciones del robot manipulador en función de los parámetros geométricos e inerciales, que caracterizan a los mecanismos o elementos espaciales, los cuales constituyen la estructura mecánica del robot, en cadena cinemática abierta.

Existen varios métodos para obtener la descripción matemática de la dinámica de un mecanismo articulado, entre las cuales destacan: Bon-graphs, D'Alambert, Hamilton, Lagrange-Euler, Newton-Euler. Las más utilizadas son las dos últimas. La metodología de *Lagrange-Euler* nos permite obtener una ecuación diferencial estructurada cuyos términos tienen una interpretación física simple como: inercia del robot, fricción, efecto de la gravedad, fuerzas centrífugas y de Coriolis. Por su parte, la metodología de *Newton-Euler* produce una formulación recursiva de las ecuaciones diferenciales que representan el movimiento del robot, lo cual no sólo facilita su implementación, sino que representa una mayor eficiencia computacional comparado con el método de Lagrange-Euler, particularmente cuando aumenta el número de articulaciones del robot.

4.3 Formulación de Lagrange

Los sistemas dinámicos complejos se pueden modelar en forma directa mediante la *formulación de Lagrange*, que se basa en la noción de la energía total, en el movimiento de sus coordenadas generalizadas y en la aplicación de fuerzas generalizadas. Para un brazo robótico de n grados de libertad (de n -ejes), un conjunto

apropiado de coordenadas generalizadas es el vector articular q de dimensión n . Los n componentes de q representan ángulos si las articulaciones son de revolución o distancias si las articulaciones son de traslación.

Para deducir las ecuaciones dinámicas del robot se hace uso del método de *Lagrange-Euler*, en paralelo con la representación matricial de *Denavit-Hartenberg*, que describen el desplazamiento espacial entre los sistemas de coordenadas de los elementos adyacentes, y así se obtiene la información cinemática de cada elemento, consiguiendo como resultado una descripción algorítmica conveniente y compacta de las ecuaciones del movimiento dinámico del robot, expresadas mediante operaciones matriciales.

El uso de este método permite diseñar las ecuaciones apropiadas, para describir dinámicamente a un robot, obteniendo modelos de robots multiejes complejos en una forma relativamente simple, elegante y transparente, así como el proceso de su realización en el computador.

La función de *Lagrange* $L(q, \dot{q})$ se define como la diferencia entre la energía cinética total $E_c(q, \dot{q})$ de un sistema y su energía potencial total $E_p(q)$ como sigue:

$$L(q(t), \dot{q}(t)) \triangleq E_c(q(t), \dot{q}(t)) - E_p(q(t)) \quad (4.1)$$

Se debe notar que la energía cinética depende tanto del vector posicional de articulaciones q como del vector velocidad de articulación \dot{q} mientras que la energía potencial depende solamente del vector posicional q .

Las energías de los elementos del robot, se deben expresar en un sistema de coordenadas apropiado e inherente al robot, de tal manera que simplifique al problema, asociando las coordenadas generalizadas a las articulaciones del robot,

para describir completamente la localización y el movimiento de cada elemento, con respecto a un sistema de coordenadas de referencia.

La ecuación general del movimiento dinámico de un robot, se formula en términos de la función de *Lagrange*, de la siguiente manera :

$$\frac{d}{dt} \frac{\partial}{\partial \dot{q}_k} L(q, \dot{q}) - \frac{\partial}{\partial q_k} L(q, \dot{q}) = F_k ; \quad 1 \leq k \leq n \quad (4.2)$$

donde:

- F_k = fuerza o torque generalizado que actúa sobre la articulación k , para mover al elemento k .
- q_k = coordenada generalizada o variable de articulación (θ_k ó d_k).
- \dot{q}_k = derivada de la coordenada generalizada k o velocidad de la articulación k .

La formulación de *Lagrange* para evaluar la dinámica de un robot expresado por la ecuación (4.2) consiste de un sistema de n ecuaciones diferenciables no lineales de segundo orden respecto del vector articular q . Para especificar estas ecuaciones en una forma mas detallada, ampliaremos los conceptos de las energías cinética y potencial y la fuerza generalizada.

4.3.1 Energía cinética

La energía cinética $E_c(q, \dot{q})$ es el término más complejo de la función de *Lagrange* para un manipulador. Para obtener una expresión de la energía cinética total del robot, primero se debe determinar la contribución de su k -ésimo eslabón a tal energía. El k -ésimo enlace o eslabón se mueve en el espacio tridimensional con una velocidad lineal y una velocidad angular tal como se aprecia en la Fig. (4.1).

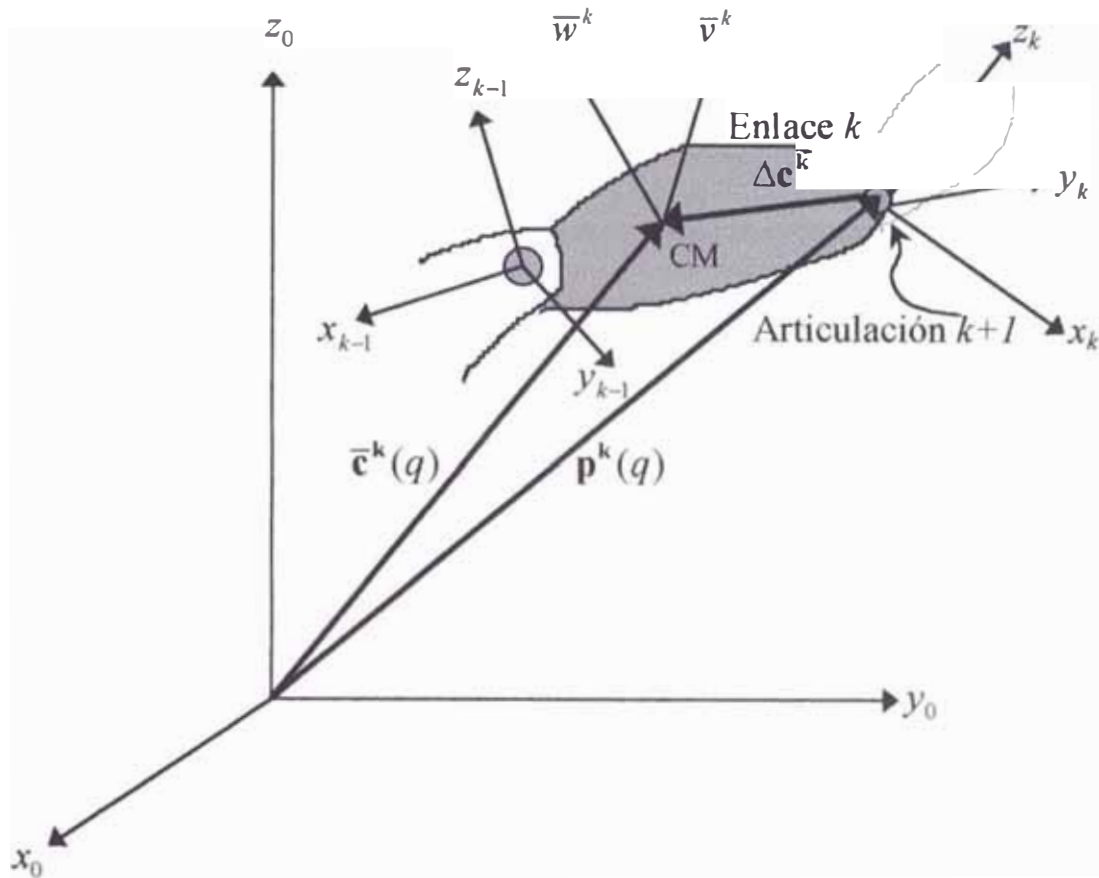


Fig. 4.1 Movimiento del enlace k de una cadena articulada.

Sean $v^k, w^k \in \mathbf{R}^3$ la velocidad lineal del centro de masa del k -ésimo eslabón y la velocidad angular relativa a su centro de masa, respectivamente, expresadas ambas con respecto al referencial base L_0 del robot. La energía cinética debido al movimiento lineal y angular del k -ésimo eslabón está representada por:

$$E_c^k(q, \dot{q}) = \frac{1}{2} [(\bar{v}^k(q, \dot{q}))^T m_k \bar{v}^k(q, \dot{q}) + (\bar{w}^k(q, \dot{q}))^T D_k(q) \bar{w}^k(q, \dot{q})] \quad (4.3)$$

donde m_k (escalar) representa la masa del k -ésimo enlace y $D_k \in \mathbf{R}^{3 \times 3}$ es el tensor de inercia del k -ésimo enlace, relativo de su centro de masa expresado con respecto al referencial base. Para determinar la energía cinética total del robot, se suman las contribuciones de la energía cinética de cada elemento k como sigue:

$$E_c(q, \dot{q}) = \sum_{k=1}^n E_c^k(q, \dot{q}) \quad (4.4)$$

Para poder formular las ecuaciones de movimiento del robot mediante la ecuación (4.2), se requiere que la energía cinética esté expresada como una función explícita de las variables articulares q y de sus velocidades \dot{q} .

En la expresión de la energía cinética, dada por la ecuación (4.3), el tensor de inercia \mathbf{D}_k del eslabón k , relativo a su centro de masa y está expresado con respecto del referencial base \mathbf{L}_0 del robot. Es decir \mathbf{D}_k es el tensor de inercia que se obtiene al trasladar el referencial \mathbf{L}_0 al centro de masa del k -ésimo eslabón para aplicar la ecuación (C.7) del anexo C.

Recordando que el algoritmo de *Denavit-Hartenberg* asigna un sistema de coordenadas o referencial \mathbf{L}_k a cada eslabón, de modo que \mathbf{L}_k y \mathbf{L}_0 se relacionan por una matriz de transformación homogénea compuesta ${}_0\mathbf{T}^k(q)$ como sigue:

$${}_0\mathbf{T}^k(q) = \begin{bmatrix} {}_0\mathbf{R}^k(q) & \mathbf{P}^k(q) \\ \mathbf{0}^T & 1 \end{bmatrix}; \quad 1 \leq k < n \quad (4.5)$$

La matriz de rotación ${}_0\mathbf{R}^k(q)$ representa la orientación del referencial \mathbf{L}_k con respecto del referencial \mathbf{L}_0 , mientras que el vector de traslación $\mathbf{p}^k(q)$ representa la posición del origen del referencial \mathbf{L}_k con respecto al referencial \mathbf{L}_0 . La energía cinética debido a la velocidad angular del enlace k con respecto a su centro de masa está expresada en la ecuación (4.3) en términos del referencial base \mathbf{L}_0 . Para reescribir esta energía en términos del referencial \mathbf{L}_k se utiliza la inversa de la matriz ${}_0\mathbf{R}^k(q)$, en decir ${}_k\mathbf{R}^0(q) = ({}_0\mathbf{R}^k(q))^{-1} = ({}_0\mathbf{R}^k(q))^T$ con la finalidad de convertir la velocidad angular w^k expresada respecto al referencial \mathbf{L}_0 al referencial \mathbf{L}_k .

Llevando a una expresión equivalente para la energía cinética del k -ésimo eslabón, debida a su velocidad angular con respecto a su centro de masa se llega es:

$$\begin{aligned}\frac{1}{2}(\bar{w}^k)^T \mathbf{D}_k \bar{w}^k &= \frac{1}{2}({}_k \mathbf{R}^0 \bar{w}^k)^T \bar{\mathbf{D}}_k ({}_k \mathbf{R}^0 \bar{w}^k) \\ \frac{1}{2}(\bar{w}^k)^T \mathbf{D}_k \bar{w}^k &= \frac{1}{2}(\bar{w}^k)^T ({}_k \mathbf{R}^0)^T \bar{\mathbf{D}}_k ({}_k \mathbf{R}^0) \bar{w}^k \\ \frac{1}{2}(\bar{w}^k)^T \mathbf{D}_k \bar{w}^k &= \frac{1}{2}(\bar{w}^k)^T {}_0 \mathbf{R}^k \bar{\mathbf{D}}_k ({}_0 \mathbf{R}^k)^T \bar{w}^k\end{aligned}\quad (4.6)$$

donde \mathbf{D}_k denota el tensor de inercia del k -ésimo enlace relativo a su centro de masa, expresado con respecto del referencial \mathbf{L}_k . Es decir \mathbf{D}_k se obtiene al realizar una traslación de los ejes del referencial \mathbf{L}_k al centro de masa del elemento k y luego se aplica la ecuación (C.7) de Apéndice C. Como el referencial \mathbf{L}_k está fijo al eslabón k y rota junto a éste, entonces el tensor de inercia \mathbf{D}_k es constante. El extremo de la izquierda en la ecuación (4.6) representa la energía cinética del k -ésimo eslabón debido a su velocidad angular, representada en las coordenadas del referencial base \mathbf{L}_0 , mientras que el extremo de la derecha en la misma ecuación, representa la misma energía cinética expresada en términos de las coordenadas del referencial \mathbf{L}_k , entonces manteniendo la velocidad angular w^k en ambos extremos de la ecuación (4.6) se determina que:

$$\mathbf{D}_k(q) = {}_0 \mathbf{R}^k(q) \bar{\mathbf{D}}_k [{}_0 \mathbf{R}^k(q)]^T \quad (4.7)$$

De esta manera \mathbf{D}_k depende del vector q a través de la matriz de rotación ${}_0 \mathbf{R}^k(q)$ de la ecuación (4.5), en donde $({}_0 \mathbf{R}^k(q))^T$ realiza la transformación desde el referencial \mathbf{L}_0 al referencial \mathbf{L}_k , mientras que ${}_0 \mathbf{R}^k(q)$ realiza la transformación desde el referencial \mathbf{L}_k de retorno al referencial \mathbf{L}_0 .

Para calcular el tensor de inercia \mathbf{D}_k del eslabón k , primero se evalúa el tensor de inercia con respecto a un referencial que se obtiene, por la traslación del referencial \mathbf{L}_k desde el extremo del k -ésimo enlace hasta su centro de masa. Si se aplica el algoritmo $D-H$ de la sección (2.3.1) adecuada y cuidadosamente, se puede elegir el referencial \mathbf{L}_k de tal manera que \mathbf{D}_k sea diagonal. Una vez obtenida esta matriz \mathbf{D}_k sea o no diagonal la cual siempre es constante, se obtiene el tensor de inercia \mathbf{D}_k del k -ésimo eslabón con respecto al referencial \mathbf{L}_0 , aplicando la ecuación (4.7).

- **Jacobiano del eslabón**

Para desarrollar una formulación explícita de la energía cinética total del robot, también se requiere expresar las velocidades $\bar{\mathbf{v}}^k$ y $\bar{\mathbf{w}}^k$ de la ecuación (4.3) en términos de q y \dot{q} , y para ello, se utilizará la matriz jacobiana $\mathbf{J}(q)$ del manipulador, la cual relaciona desplazamientos infinitesimales de las variables articulares con los desplazamientos infinitesimales (lineales y angulares) del efector final, o bien evaluar las velocidades articulares instantáneas y transformarlas en velocidades instantáneas, lineal y angular del efector final respecto del referencial base \mathbf{L}_0 y se expresa de la siguiente manera:

$$\begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\phi} \end{bmatrix} = \mathbf{J}(q)\dot{q} \quad (4.8)$$

Para expresar $\bar{\mathbf{v}}^k$ y $\bar{\mathbf{w}}^k$ en términos de q y \dot{q} se requiere, de manera análoga encontrar la matriz jacobiana del k -ésimo eslabón, denotada por $\mathbf{J}^k(q)$, considerando que el centro de masa del eslabón k como si fuera el extremo del órgano terminal, entonces podemos obtener:

$$\begin{bmatrix} \bar{\mathbf{v}}^k \\ \bar{\mathbf{w}}^k \end{bmatrix} = \mathbf{J}^k(q)\dot{q}; \quad 1 \leq k \leq n \quad (4.9)$$

De la figura (4.1) se aprecia que $\bar{\mathbf{c}}^k \in \mathbf{R}^3$ denota el vector posicional del centro de masa del k -ésimo eslabón, expresado en el referencial \mathbf{L}_0 , y $\Delta\mathbf{c}^k \in \mathbf{R}^3$ denota el vector de desplazamiento del centro de masa del k -ésimo enlace, expresado en el referencial \mathbf{L}_k , el cual está fijo en un punto del enlace k . El vector $\Delta\mathbf{c}^k$ es constante y depende físicamente de la forma y tamaño del eslabón. La posición del centro de masa del enlace k también puede ser expresado en el referencial de la base \mathbf{L}_0 , transformando las coordenadas del vector $\Delta\mathbf{c}^k$ mediante la matriz de rotación de paso generalizado ${}^0\mathbf{R}^k(q)$ y sumándole el desplazamiento $\mathbf{p}^k(q)$, que viene a ser el origen del referencial \mathbf{L}_k , como se puede apreciar en la Fig. (4.1) y se obtiene el vector \mathbf{c}^k matemáticamente como:

$$\bar{\mathbf{c}}^k(q) = {}^0\mathbf{R}^k(q)\Delta\mathbf{c}^k + \mathbf{p}^k(q); \quad 1 \leq k \leq n \quad (4.10)$$

Una vez que ha sido localizado el centro de masa del eslabón k , la matriz jacobiana $\mathbf{J}^k(q)$ del eslabón k de $6 \times n$ puede ser formulada con respecto a este punto de referencia de la siguiente manera:

$$\mathbf{J}^k(q) = \left[\begin{array}{ccc|c} \frac{\partial \bar{\mathbf{c}}^k}{\partial q_1} & \dots & \frac{\partial \bar{\mathbf{c}}^k}{\partial q_k} & 0 \\ \hline \xi_1 z^0 & \dots & \xi_k z^{k-1} & 0 \end{array} \right] \Delta \begin{bmatrix} J_v^k(q) \\ J_w^k(q) \end{bmatrix} \quad (4.11)$$

El escalar ξ_i es un parámetro que indica el tipo de la i -ésima articulación, toma el valor de 1 si la articulación i es de revolución o 0 si la articulación es prismática o traslacional. El vector z^i representa el tercer vector unitario del referencial \mathbf{L}_i con

respecto al referencial base L_0 que se obtiene de la siguiente manera:

$$z^i(q) = {}_0\mathbf{R}^i(q) z^{i-1}; \quad 0 \leq i \leq k \quad (4.12)$$

Hay que notar que las últimas $n-k$ columnas de $\mathbf{J}^k(q)$ son iguales a cero, debido a que el movimiento del eslabón k no se ve afectado por las variables articulares distales $\{q_{k+1}, \dots, q_n\}$. El lado derecho de la ecuación (4.11) se ha particionado en dos submatrices jacobianas lineal $J_V^k(q)$ y angular $J_W^k(q)$.

- **Tensor de inercia del manipulador**

Si se usan las submatrices particionadas $J_V^k(q)$ y $J_W^k(q)$ de la matriz jacobiana $\mathbf{J}^k(q)$ del eslabón k , de la ecuación (4.11) para descomponer en las dos siguientes ecuaciones separadas, entonces la ecuación (4.11) se puede expresar como:

$$\bar{v}^k(q, \dot{q}) = J_V^k(q) \dot{q}; \quad 1 \leq k \leq n \quad (4.13.a)$$

$$\bar{w}^k(q, \dot{q}) = J_W^k(q) \dot{q}; \quad 1 \leq k \leq n \quad (4.13.b)$$

La expresión (4.13.a) especifica la velocidad lineal del centro de masa del eslabón k , mientras que la ecuación (4.13.b) especifica la velocidad angular relativa a su centro de masa. Ahora ya es posible desarrollar una expresión explícita para la energía cinética total del robot en función de las posiciones q y sus velocidades articulares \dot{q} . Si se sustituyen las ecuaciones (4.13) en la expresión original para la energía cinética total del robot dada por la ecuación (4.4), resulta:

$$E_c(q, \dot{q}) = \frac{1}{2} \sum_{k=1}^n [(\bar{v}^k)^T m_k \bar{v}^k + (\bar{w}^k)^T D_k \bar{w}^k]$$

$$E_c(q, \dot{q}) = \frac{1}{2} \sum_{k=1}^n [(J_V^k \dot{q})^T m_k J_V^k \dot{q} + (J_W^k \dot{q})^T D_k J_W^k \dot{q}]$$

$$E_c(q, \dot{q}) = \frac{1}{2} \sum_{k=1}^n [\dot{q}^T (J_V^k)^T m_k J_V^k \dot{q} + \dot{q}^T (J_W^k)^T D_k J_W^k \dot{q}]$$

$$E_c(q, \dot{q}) = \frac{1}{2} \dot{q}^T \sum_{k=1}^n [(J_V^k)^T m_k J_V^k + (J_W^k)^T D_k J_W^k] \dot{q} \quad (4.14)$$

En la expresión (4.14), tanto las submatrices del Jacobiano $J_V^k(q)$, $J_W^k(q) \in \mathbf{R}^{3 \times n}$, así como el tensor de inercia del eslabón $\mathbf{D}_k(q) \in \mathbf{R}^{3 \times 3}$, dependen del vector de coordenadas articulares q . Esta expresión para la energía cinética del robot se puede simplificar enormemente introduciendo la matriz $\mathbf{D}^k(q)$ de $n \times n$ que representa el tensor de inercia debido al movimiento traslacional y rotacional del k -ésimo elemento como:

$$\mathbf{D}^k(q) \triangleq [J_V^k(q)]^T m_k J_V^k(q) + [J_W^k(q)]^T D_k J_W^k(q) \quad (4.15)$$

Entonces, si se suman los tensores de inercia $\mathbf{D}^k(q)$ de cada eslabón k individual, se obtiene el tensor de inercia del manipulador como:

$$\mathbf{D}(q) = \sum_{k=1}^n \mathbf{D}^k(q) \quad (4.16)$$

La matriz $\mathbf{D}(q) \in \mathbf{R}^{n \times n}$, denominada tensor de inercia del robot, la cual es una matriz simétrica y definida positiva. Sustituyendo la ecuación (4.16) en la ecuación (4.14), la energía cinética total del robot queda expresada en términos del tensor de inercia del manipulador y del vector velocidad de articulación como sigue:

$$E_c(q, \dot{q}) = \frac{1}{2} \dot{q}^T \mathbf{D}(q) \dot{q} \quad (4.17)$$

De la ecuación (4.17) se puede probar que la energía cinética satisface $E_c(q, \dot{q}) \geq 0$ y que $E_c(q, \dot{q}) = 0$ si y sólo si $\dot{q} = 0$, lo cual corresponde al robot en reposo.

4.3.2 Energía potencial

Para completar la *formulación de Lagrange* de un manipulador robótico, falta analizar la energía potencial total del manipulador, denotada por $E_p(q)$. Así pues, la energía potencial almacenada en el k -ésimo eslabón del robot, es la cantidad de trabajo requerido para desplazar el centro de masa del eslabón k desde un plano de referencia horizontal en presencia de la gravedad. Sea $g \in \mathbf{R}^3$ el vector que denota la aceleración gravitacional, con respecto al referencial base \mathbf{L}_0 , el cual es un sistema de referencia inercial. Entonces en coordenadas del referencial base, el trabajo requerido $u_k(q)$ para desplazar el eslabón k a la posición $\mathbf{c}^k(q)$ es:

$$u_k(q) = -m_k \mathbf{g}^T \bar{\mathbf{c}}^k(q) \quad (4.18)$$

La energía potencial total almacenada en los eslabones del robot será la suma de las energías potenciales almacenadas en los eslabones y se expresa como:

$$E_p(q) = -\sum_{k=1}^n u_k(q) \quad (4.19)$$

Para todo robot que esté colocado sobre una base horizontal, sobre el piso o una mesa, el vector de aceleración gravitacional tiene una sola componente (sobre el vector unitario z) cuya magnitud es de 9.8062 m/s^2 y sentido negativo (en dirección al centro de la tierra).

Para simplificar la expresión final para la energía potencial, se introduce una nueva cantidad $c(q)$ que se define como la suma ponderada de los centros de masa de los n eslabones del robot por sus masas respectivas, y se expresa como:

$$\bar{\mathbf{c}}(q) \triangleq \sum_{k=1}^n m_k \mathbf{c}^k(q) \quad (4.20)$$

Combinando las ecuaciones (4.18), (4.19) y (4.20), la expresión de la energía potencial del robot se expresa en términos de la suma ponderada $c(q)$ como sigue:

$$E_p(q) = -g^T \bar{c}(q) \quad (4.21)$$

Nótese que $E_p(q)$ es máxima cuando todos los eslabones del robot están opuestos directamente a g , nula cuando todos los eslabones son ortogonales a g y mínima cuando todos los eslabones están alineados con g .

4.3.3 Fuerzas generalizadas

En base a las expresiones simplificadas correspondientes a las energías cinética y potencial del robot, se puede escribir el *Lagrangeano* del manipulador, combinando las ecuaciones (4.1), (4.17) y (4.21) de la siguiente manera:

$$L(q, \dot{q}) = \frac{1}{2} \dot{q}^T \mathbf{D}(q) \dot{q} + g^T \bar{c}(q) \quad (4.22)$$

Lo que queda a continuación es derivar a $L(q, \dot{q})$ con respecto a q , \dot{q} y a t para formular las ecuaciones dinámicas del movimiento del robot. Sin embargo antes de hacer esto primero se debe desarrollar una formulación más detallada de los términos que aparecen en el lado derecho de la ecuación (4.2) que viene ha ser la fuerza generalizada k aplicada a la k -ésima articulación del manipulador expresada por la ecuación (C.6) del anexo C como:

$$F_k = \tau_k - b_k(\dot{q}); \quad 1 \leq k < n \quad (4.23)$$

donde τ_k es el torque articular generado por el actuador k . La expresión de la fuerza de fricción $b_k(\dot{q})$ para la articulación k está formulada en la ecuación (C.3) del Anexo C como:

$$b_k(\dot{q}) = b_k^v \dot{q}_k + \operatorname{sgn}(\dot{q}_k) \left[b_k^d + (b_k^s - b_k^v) \exp\left(\frac{-|\dot{q}_k|}{\varepsilon}\right) \right]; \quad 1 \leq k < n \quad (4.24)$$

donde sgn denota la función signo. El término de la fricción en la ecuación (4.23) tiene signo negativo porque la fuerza de la fricción se opone al movimiento generada por los actuadores.

Es posible también sumar un tercer término al vector fuerza generalizado, debido a los torques de las articulaciones inducidas por la fuerza del efector final y el vector momento F^{herr} . Sin embargo, para mantener un modelo dinámico simple asumiremos que el brazo se mueve libremente en su espacio de trabajo y que cualquier carga que esté cargando, está implícitamente incluida en la descripción física del último eslabón. Bajo estas condiciones, las fuerzas del efector final y el vector momento son nulas.

4.3.4 Modelo dinámico de Lagrange-Euler

Ahora que se dispone de expresiones detalladas para la energía cinética y la energía potencial, así como para las fuerzas generalizadas disponibles, ya se puede aplicar la ecuación de *Lagrange* dada por la ecuación (4.2) para desarrollar un modelo dinámico general de un robot. Para facilitar el cálculo de las derivadas requeridas, se puede expandir la fórmula de la energía cinética de la ecuación (4.17) y de la energía potencial en la ecuación (4.21) en suma de escalares, como sigue:

$$E_c(q, \dot{q}) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \mathbf{D}_{ij}(q) \dot{q}_i \dot{q}_j \quad (4.25)$$

$$E_p(q) = - \sum_{i=1}^3 \sum_{j=1}^n g_i m_j c_i^j(q) \quad (4.26)$$

Sustituyendo las ecuaciones (4.25) y (4.26) en la ecuación (4.1) se obtiene una expresión de *Lagrange* en función de las variables articulares y sus velocidades como sigue:

$$L(q, \dot{q}) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \mathbf{D}_{ij}(q) \dot{q}_i \dot{q}_j + \sum_{i=1}^3 \sum_{j=1}^n g_i m_j \bar{c}_i^j(q) \quad (4.27)$$

La matriz $\mathbf{D}(q)$ es una matriz simétrica y definida positiva, la derivada del *Lagrangeano* de la ecuación (4.27) con respecto de \dot{q}_k se obtiene de la siguiente manera:

$$\begin{aligned} \frac{\partial}{\partial \dot{q}_k} L(q, \dot{q}) &= \frac{\partial}{\partial \dot{q}_k} E_c(q, \dot{q}) = \frac{1}{2} \frac{\partial}{\partial \dot{q}_k} \sum_{i=1}^n \sum_{j=1}^n \mathbf{D}_{ij}(q) \dot{q}_i \dot{q}_j \\ \frac{\partial}{\partial \dot{q}_k} L(q, \dot{q}) &= \frac{1}{2} \frac{\partial}{\partial \dot{q}_k} \left[\mathbf{D}_{kk}(q) \dot{q}_k^2 + \dot{q}_k \sum_{j \neq k} \mathbf{D}_{kj}(q) \dot{q}_j + \dot{q}_k \sum_{i \neq k} \mathbf{D}_{ik}(q) \dot{q}_i \right] \\ \frac{\partial}{\partial \dot{q}_k} L(q, \dot{q}) &= \frac{1}{2} \left[2\mathbf{D}_{kk}(q) \dot{q}_k + \sum_{j \neq k} \mathbf{D}_{kj}(q) \dot{q}_j + \sum_{i \neq k} \mathbf{D}_{ik}(q) \dot{q}_i \right] \\ \frac{\partial}{\partial \dot{q}_k} L(q, \dot{q}) &= \sum_{j=1}^n \mathbf{D}_{kj}(q) \dot{q}_j \end{aligned} \quad (4.28)$$

Derivando la ecuación (4.28) con respecto al tiempo se logra obtener el primer término de la formulación de *Lagrange* aplicado a un manipulador de la siguiente manera:

$$\begin{aligned} \frac{d}{dt} \frac{\partial}{\partial \dot{q}_k} L(q, \dot{q}) &= \frac{d}{dt} \sum_{j=1}^n \mathbf{D}_{kj}(q) \dot{q}_j \\ \frac{d}{dt} \frac{\partial}{\partial \dot{q}_k} L(q, \dot{q}) &= \sum_{j=1}^n \mathbf{D}_{kj}(q) \ddot{q}_j + \sum_{j=1}^n \left[\frac{d}{dt} \mathbf{D}_{kj}(q) \right] \dot{q}_j \end{aligned}$$

$$\frac{d}{dt} \frac{\partial}{\partial \dot{q}_k} L(q, \dot{q}) = \sum_{j=1}^n \mathbf{D}_{kj}(q) \ddot{q}_j + \sum_{j=1}^n \left\{ \sum_{i=1}^n \left[\frac{\partial}{\partial q_i} \mathbf{D}_{kj}(q) \right] \dot{q}_i \right\} \dot{q}_j$$

$$\frac{d}{dt} \frac{\partial}{\partial \dot{q}_k} L(q, \dot{q}) = \sum_{j=1}^n \mathbf{D}_{kj}(q) \ddot{q}_j + \sum_{i=1}^n \sum_{j=1}^n \left[\frac{\partial}{\partial q_i} \mathbf{D}_{kj}(q) \right] \dot{q}_i \dot{q}_j \quad (4.29)$$

Derivando el *Lagrangiano* de la ecuación (4.27) con respecto a la k -ésima variable de articulación q_k , se obtiene el segundo término de la ecuación diferencial de *Lagrange* como sigue:

$$\frac{\partial}{\partial q_k} L(q, \dot{q}) = \frac{\partial}{\partial q_k} E_c(q, \dot{q}) - \frac{\partial}{\partial q_k} E_p(q)$$

$$\frac{\partial}{\partial q_k} L(q, \dot{q}) = \frac{1}{2} \frac{\partial}{\partial q_k} \sum_{i=1}^n \sum_{j=1}^n \mathbf{D}_{ij}(q) \dot{q}_i \dot{q}_j + \frac{\partial}{\partial q_k} \sum_{i=1}^3 \sum_{j=1}^n g_i m_j \bar{c}_i^j(q)$$

$$\frac{\partial}{\partial q_k} L(q, \dot{q}) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \left[\frac{\partial}{\partial q_k} \mathbf{D}_{ij}(q) \right] \dot{q}_i \dot{q}_j + \sum_{i=1}^3 \sum_{j=1}^n g_i m_j \left[\frac{\partial}{\partial q_k} \bar{c}_i^j(q) \right]$$

$$\frac{\partial}{\partial q_k} L(q, \dot{q}) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \left[\frac{\partial}{\partial q_k} \mathbf{D}_{ij}(q) \right] \dot{q}_i \dot{q}_j + \sum_{i=1}^3 \sum_{j=1}^n g_i m_j J_{V_{ik}}^j$$

$$\frac{\partial}{\partial q_k} L(q, \dot{q}) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \left[\frac{\partial}{\partial q_k} \mathbf{D}_{ij}(q) \right] \dot{q}_i \dot{q}_j + \sum_{i=1}^3 \sum_{j=k}^n g_i m_j J_{V_{ik}}^j \quad (4.30)$$

El índice en la última sumatoria de la ecuación (4.30) va desde $j = k$ en vez de $j = 1$, debido que de la ecuación (4.11) las últimas $n-j$ columnas de J_V^j son cero. Para simplificar la ecuación final de movimiento se introducen dos nuevos términos: el vector $h(q) \in \mathbf{R}^n$ definido como carga debida al efecto gravitatorio y la matriz $\mathbf{C}^k \in \mathbf{R}^{n \times n}$ denominada matriz de acoplamiento de velocidades para la k -ésima articulación. Ambas cantidades se expresan matemáticamente como:

$$h_k(q) \triangleq - \sum_{i=1}^3 \sum_{j=i}^n g_i m_j J_{V_{ik}}^j(q); \quad 1 \leq k \leq n \quad (4.31)$$

$$C_{ij}^k(q) \triangleq \frac{\partial}{\partial q_k} \mathbf{D}_{kj}(q) - \frac{1}{2} \frac{\partial}{\partial q_k} \mathbf{D}_{ij}(q); \quad 1 \leq i, j, k \leq n \quad (4.32)$$

Sustituyendo las ecuaciones (4.23), (4.29) y (4.30) en la función de *Lagrange* expresada por la ecuación (4.2) y usando las ecuaciones (4.31) y (4.32), se obtiene una formulación diferencial y concisa del modelo dinámico del movimiento del robot. Por tanto la expresión para el torque generado por el *k*-ésimo actuador para mover a la articulación *k* es como sigue:

$$\begin{aligned} \tau_k &= F_k + b_k(\dot{q}) \\ \tau_k &= \frac{d}{dt} \frac{\partial}{\partial \dot{q}_k} L(q, \dot{q}) - \frac{\partial}{\partial q_k} L(q, \dot{q}) + b_k(\dot{q}) \\ \tau_k &= \sum_{j=1}^n \mathbf{D}_{kj}(q) \ddot{q}_j + \sum_{i=1}^n \sum_{j=1}^n \left[\frac{\partial}{\partial q_i} \mathbf{D}_{kj}(q) \right] \dot{q}_i \dot{q}_j - \frac{\partial}{\partial q_k} L(q, \dot{q}) + b_k(\dot{q}) \\ \tau_k &= \sum_{j=1}^n \mathbf{D}_{kj}(q) \ddot{q}_j + \sum_{i=1}^n \sum_{j=1}^n \left[\frac{\partial}{\partial q_k} \mathbf{D}_{kj}(q) - \frac{1}{2} \frac{\partial}{\partial q_k} \mathbf{D}_{ij}(q) \right] \dot{q}_i \dot{q}_j + h_k(q) + b_k(\dot{q}) \\ \tau_k &= \sum_{j=1}^n \mathbf{D}_{kj}(q) \ddot{q}_j + \sum_{i=1}^n \sum_{j=1}^n C_{ij}^k \dot{q}_i \dot{q}_j + h_k(q) + b_k(\dot{q}) \end{aligned} \quad (4.33)$$

El primer término de la ecuación (4.33), es un término aceleración que representa las fuerzas inerciales generadas por el movimiento de los enlaces del robot. El segundo término es un término del producto de las velocidades asociado con las fuerzas de Coriolis y centrífugas. El tercer término es un término de posición que representa la carga debido a la gravedad. Finalmente, el cuarto término es un término velocidad que representa la fricción que se opone al movimiento del robot.

El término del producto de velocidades articulares de la ecuación (4.33), se puede escribir como una función cuadrática de la siguiente manera:

$$c_k(q, \dot{q}) \triangleq \dot{q}^T C^k \dot{q}; \quad 1 < k \leq n \quad (4.34)$$

El vector de acoplamiento de velocidades $c_k(q, \dot{q})$ se puede expandir mediante dos tipos distintos de acoplamiento de velocidades que puedan aparecer entre ejes y se expresa de la siguiente manera:

$$c_k(q, \dot{q}) = \sum_{i=1}^n C_{ii}^k(q) \dot{q}_i^2 + \sum_{i=1}^n \sum_{j \neq i} C_{ij}^k(q) \dot{q}_i \dot{q}_j \quad (4.35)$$

Las dos sumatorias en la ecuación (4.35) se generan por los componentes de la diagonal y fuera de la diagonal de la matriz de acoplamiento respectivamente. La primera sumatoria corresponde a los términos del cuadrado de la velocidad asociada con la fuerza centrífuga. En este caso el término $C_{ii}^k(q) \dot{q}_i^2$ representa la fuerza centrífuga actuando sobre la articulación k debido al movimiento de la articulación i .

La segunda sumatoria en la ecuación (4.35) corresponde a los términos del producto de velocidades asociadas con la fuerza de Coriolis. En este caso el término $C_{ij}^k(q) \dot{q}_i \dot{q}_j$ representa la fuerza de Coriolis actuando sobre la articulación k debido a la combinación de movimientos de las articulaciones i y j cuando $j \neq i$.

De la ecuación (4.35) se puede ver claramente que el vector de acoplamiento de velocidades es nulo cuando el vector velocidad es cero, es decir $c(q, 0) = 0$ y se puede interpretar también despreciable cuando el robot está moviéndose a bajas velocidades.

Escribiendo el término $c_k(q, \dot{q})$ de la ecuación (4.34) en la ecuación (4.33) se podrían combinar n ecuaciones diferenciales, escalares y simultáneas para un

manipulador de n grados de libertad que se encuentra moviéndose libremente en su espacio o volumen de trabajo por efecto de los torques τ generados por sus motores que actúan sobre las articulaciones del robot, obteniendo de este modo un modelo dinámico para el brazo robótico mediante una expresión vectorial compacta y simple como sigue:

$$\tau(t) = D(q(t)) \ddot{q}(t) + c(q(t), \dot{q}(t)) + h(q(t)) + b(\dot{q}(t)) \quad (4.36)$$

donde:

- $\tau(t)$ = vector fuerza o torque que actúa sobre cada una de las articulaciones del robot .
- $D(q)$ = matriz simétrica no singular $n \times n$ que relaciona el efecto inercial de todos los elementos, que expresa la fuerza inercial debido al movimiento del robot.
- $c(q, \dot{q})$ = vector de fuerzas generado por la reacción de los efectos centrífugo y de Corioles del robot, debido al acoplamiento de velocidades durante el movimiento.
- $h(q)$ = vector fuerza, debido a la presencia de la gravedad que actúa sobre los elementos del robot.
- $b(\dot{q})$ = vector de fuerzas de fricción, opuestas al movimiento de las articulaciones del robot.

4.4 Dinámica del *ROB-DEX*

Para evaluar la dinámica del *ROB-DEX* por el método de *Lagrange-Euler* se debe formular la expresión del *lagrangeano*; para esto se debe evaluar las energías cinética y potencial. Para realizar las transformaciones requeridas se harán uso de las expresiones (2.18) (2.19) (2.20) y (2.21) de la cinemática del *ROB-DEX*, entonces

multiplicando ordenadamente estas ecuaciones obtenemos las siguientes matrices de transformación homogénea:

$${}^0\mathbf{T}^1 = \begin{bmatrix} \cos q_1 & -\text{sen} q_1 & 0 & a_1 \cos q_1 \\ \text{sen} q_1 & \cos q_1 & 0 & a_1 \text{sen} q_1 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.37.a)$$

$${}^0\mathbf{T}^2 = \begin{bmatrix} \cos(q_1 + q_2) & -\text{sen}(q_1 + q_2) & 0 & a_2 \cos(q_1 + q_2) + a_1 \cos q_1 \\ \text{sen}(q_1 + q_2) & \cos(q_1 + q_2) & 0 & a_2 \text{sen}(q_1 + q_2) + a_1 \text{sen} q_1 \\ 0 & 0 & 1 & d_1 - d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.37.b)$$

$${}^0\mathbf{T}^3 = \begin{bmatrix} \cos(q_1 + q_2 + q_3) & -\text{sen}(q_1 + q_2 + q_3) & 0 & a_2 \cos(q_1 + q_2) + a_1 \cos q_1 \\ \text{sen}(q_1 + q_2 + q_3) & \cos(q_1 + q_2 + q_3) & 0 & a_2 \text{sen}(q_1 + q_2) + a_1 \text{sen} q_1 \\ 0 & 0 & 1 & d_1 - d_2 - d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.37.c)$$

$${}^0\mathbf{T}^4 = \begin{bmatrix} \cos(q_1 + q_2 + q_3) & \text{sen}(q_1 + q_2 + q_3) & 0 & a_2 \cos(q_1 + q_2) + a_1 \cos q_1 \\ \text{sen}(q_1 + q_2 + q_3) & -\cos(q_1 + q_2 + q_3) & 0 & a_2 \text{sen}(q_1 + q_2) + a_1 \text{sen} q_1 \\ 0 & 0 & -1 & d_1 - d_2 - d_3 - q_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.37.d)$$

Los elementos o eslabones que constituyen la estructura mecánica del *ROB-DEX* son heterogéneos (de diferente material) y geoméricamente complejos; es decir estos elementos están formados por diferentes materiales que adoptan diversas formas geométricas, tamaños diferentes y masas distintas, por lo que es muy difícil y hasta imposible evaluar los momentos de inercia de los elementos haciendo uso de la expresión (C.9). Debido a esto, se han obtenido las magnitudes de los tensores de inercia $\bar{\mathbf{D}}_k$ relativos al centro de masa de cada elemento con respecto al referencial \mathbf{L}_k usando métodos experimentales en forma aproximada y que se expresan respectivamente como:

$$\bar{\mathbf{D}}_1 = \begin{bmatrix} I_{xx1} & 0 & 0 \\ 0 & I_{yy1} & 0 \\ 0 & 0 & I_{zz1} \end{bmatrix} = \begin{bmatrix} 305.45 & 0 & 0 \\ 0 & 3895.12 & 0 \\ 0 & 0 & 5027.22 \end{bmatrix} \text{Kg. cm}^2 \quad (4.38.a)$$

$$\bar{\mathbf{D}}_2 = \begin{bmatrix} I_{xx2} & 0 & 0 \\ 0 & I_{yy2} & 0 \\ 0 & 0 & I_{zz2} \end{bmatrix} = \begin{bmatrix} 120.34 & 0 & 0 \\ 0 & 1342.46 & 0 \\ 0 & 0 & 110.52 \end{bmatrix} \text{Kg. cm}^2 \quad (4.38.b)$$

$$\bar{\mathbf{D}}_3 = \begin{bmatrix} I_{xx3} & 0 & 0 \\ 0 & I_{yy3} & 0 \\ 0 & 0 & I_{zz3} \end{bmatrix} = \begin{bmatrix} 154.43 & 0 & 0 \\ 0 & 158.22 & 0 \\ 0 & 0 & 28.86 \end{bmatrix} \text{Kg.cm}^2 \quad (4.38.c)$$

$$\bar{\mathbf{D}}_4 = \begin{bmatrix} I_{xx4} & 0 & 0 \\ 0 & I_{yy4} & 0 \\ 0 & 0 & I_{zz4} \end{bmatrix} = \begin{bmatrix} 751.45 & 0 & 0 \\ 0 & 743.54 & 0 \\ 0 & 0 & 1.235 \end{bmatrix} \text{Kg. cm}^2 \quad 4.38.d)$$

El tensor de inercia $\mathbf{D}_k(q)$ de cada enlace relativo a su centro de masa del elemento k expresado con respecto al referencial base \mathbf{L}_0 se determina utilizando la ecuación (4.7), combinando las ecuaciones (4.37) y (4.38).

El tensor de inercia $\mathbf{D}_1(q)$ del primer enlace respecto del referencial \mathbf{L}_0 es:

$$\mathbf{D}_1(q) = \begin{bmatrix} C_1 & -S_1 & 0 \\ S_1 & C_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} I_{xx1} & 0 & 0 \\ 0 & I_{yy1} & 0 \\ 0 & 0 & I_{zz1} \end{bmatrix} \begin{bmatrix} C_1 & S_1 & 0 \\ -S_1 & C_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{D}_1(q) = \begin{bmatrix} C_1^2 I_{xx1} + S_1^2 I_{yy1} & C_1 S_1 (I_{xx1} - I_{yy1}) & 0 \\ C_1 S_1 (I_{xx1} - I_{yy1}) & S_1^2 I_{xx1} + C_1^2 I_{yy1} & 0 \\ 0 & 0 & I_{zz1} \end{bmatrix} \quad (4.39.a)$$

El tensor de inercia $\mathbf{D}_2(q)$ del segundo enlace respecto del referencial \mathbf{L}_0 es:

$$\mathbf{D}_2(q) = \begin{bmatrix} C_{12} & -S_{12} & 0 \\ S_{12} & C_{12} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} I_{xx2} & 0 & 0 \\ 0 & I_{yy2} & 0 \\ 0 & 0 & I_{zz2} \end{bmatrix} \begin{bmatrix} C_{12} & S_{12} & 0 \\ -S_{12} & C_{12} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{D}_2(q) = \begin{bmatrix} C_{12}^2 I_{xx2} + S_{12}^2 I_{yy2} & C_{12} S_{12} (I_{xx2} - I_{yy2}) & 0 \\ C_{12} S_{12} (I_{xx2} - I_{yy2}) & S_{12}^2 I_{xx2} + C_{12}^2 I_{yy2} & 0 \\ 0 & 0 & I_{zz2} \end{bmatrix} \quad (4.39.b)$$

El tensor de inercia $\mathbf{D}_3(q)$ del tercer enlace respecto del referencial \mathbf{L}_0 es:

$$\mathbf{D}_3(q) = \begin{bmatrix} C_{123} & -S_{123} & 0 \\ S_{123} & C_{123} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} I_{xx3} & 0 & 0 \\ 0 & I_{yy3} & 0 \\ 0 & 0 & I_{zz3} \end{bmatrix} \begin{bmatrix} C_{123} & S_{123} & 0 \\ -S_{123} & C_{123} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{D}_3(q) = \begin{bmatrix} C_{123}^2 I_{xx3} + S_{123}^2 I_{yy3} & C_{123} S_{123} (I_{xx3} - I_{yy3}) & 0 \\ C_{123} S_{123} (I_{xx3} - I_{yy3}) & S_{123}^2 I_{xx3} + C_{123}^2 I_{yy3} & 0 \\ 0 & 0 & I_{zz3} \end{bmatrix} \quad (4.39.c)$$

El tensor de inercia $\mathbf{D}_4(q)$ del cuarto enlace respecto del referencial \mathbf{L}_0 es:

$$\mathbf{D}_4(q) = \begin{bmatrix} C_{123} & S_{123} & 0 \\ S_{123} & -C_{123} & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} I_{xx4} & 0 & 0 \\ 0 & I_{yy4} & 0 \\ 0 & 0 & I_{zz4} \end{bmatrix} \begin{bmatrix} C_{123} & S_{123} & 0 \\ S_{123} & -C_{123} & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

$$\mathbf{D}_4(q) = \begin{bmatrix} C_{123}^2 I_{xx4} + S_{123}^2 I_{yy4} & C_{123} S_{123} (I_{xx4} - I_{yy4}) & 0 \\ C_{123} S_{123} (I_{xx4} - I_{yy4}) & S_{123}^2 I_{xx4} + C_{123}^2 I_{yy4} & 0 \\ 0 & 0 & I_{zz4} \end{bmatrix} \quad (4.39.d)$$

Los vectores correspondientes a los centros de masa ΔC^k de los enlaces del *ROB-DEX*, con respecto a sus sistemas de referencia \mathbf{L}_k adheridos a cada enlace, matemáticamente también son difíciles de obtener y se han obtenido en forma experimental y sus valores aproximados son

$$\overline{\Delta C}^1 = \begin{pmatrix} p_{CMx1} & p_{CMy1} & p_{CMz1} \end{pmatrix}^T = (-18.0 \quad 0.0 \quad 4.5)^T \text{ cm} \quad (4.40.a)$$

$$\overline{\Delta C}^2 = \begin{pmatrix} p_{CMx2} & p_{CMy2} & p_{CMz2} \end{pmatrix}^T = (-10.20 \quad 0.0 \quad 7.4)^T \text{ cm} \quad (4.40.b)$$

$$\overline{\Delta C}^3 = \begin{pmatrix} p_{CMx3} & p_{CMy3} & p_{CMz3} \end{pmatrix}^T = (-3.00 \quad 0.0 \quad 11.5)^T \text{ cm} \quad (4.40.c)$$

$$\overline{\Delta C}^4 = \begin{pmatrix} p_{CMx4} & p_{CMy4} & p_{CMz4} \end{pmatrix}^T = (2.00 \quad 0.0 \quad -35.2)^T \text{ cm} \quad (4.40.d)$$

Los elementos del *ROB-DEX* se han diseñado de tal manera que sus centros de gravedad sean simétricos al eje x_k , con respecto al plano x_k - y_k , por lo que los puntos p_{CMyk} son todos igual a cero. De esta manera las masas de los elementos de la estructura del robot están distribuidas simétricamente con respecto al eje x_k .

El centro de masa $\bar{c}^k(q)$ de cada elemento del *ROB-DEX* expresado con respecto al referencial base L_0 se puede formular usando la ecuación (4.10), combinando las ecuaciones (4.37) y (4.40).

El centro de masa $\bar{c}^1(q)$ del primer elemento respecto del referencial L_0 es:

$$\bar{c}^1(q) = \begin{bmatrix} C_1 & -S_1 & 0 \\ S_1 & C_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_{CMx1} \\ p_{CMy1} \\ p_{CMz1} \end{bmatrix} + \begin{bmatrix} a_1 \cos q_1 \\ a_1 \operatorname{sen} q_1 \\ d_1 \end{bmatrix}$$

$$\bar{c}^1(q) = \begin{bmatrix} C_1 p_{CMx1} + a_1 \cos q_1 \\ S_1 p_{CMx1} + a_1 \operatorname{sen} q_1 \\ p_{CMz1} + d_1 \end{bmatrix} \quad (4.41.a)$$

El centro de masa $c^2(q)$ del segundo elemento respecto del referencial L_0 es:

$$\bar{\mathbf{c}}^2(q) = \begin{bmatrix} C_{12} & -S_{12} & 0 \\ S_{12} & C_{12} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_{CMx2} \\ p_{CMy2} \\ p_{CMz2} \end{bmatrix} + \begin{bmatrix} a_1 C_1 + a_2 C_{12} \\ a_1 S_1 + a_2 S_{12} \\ d_1 - d_2 \end{bmatrix}$$

$$\bar{\mathbf{c}}^2(q) = \begin{bmatrix} C_{12} p_{CMx2} + a_1 C_1 + a_2 C_{12} \\ S_{12} p_{CMx2} + a_1 S_1 + a_2 S_{12} \\ p_{CMz2} + d_1 - d_2 \end{bmatrix} \quad (4.41.b)$$

El centro de masa $\bar{\mathbf{c}}^3(q)$ del tercer elemento respecto del referencial \mathbf{L}_0 es:

$$\bar{\mathbf{c}}^3(q) = \begin{bmatrix} C_{123} & -S_{123} & 0 \\ S_{123} & C_{123} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_{CMx3} \\ p_{CMy3} \\ p_{CMz3} \end{bmatrix} + \begin{bmatrix} a_1 C_1 + a_2 C_{12} \\ a_1 S_1 + a_2 S_{12} \\ d_1 - d_2 - d_3 \end{bmatrix}$$

$$\bar{\mathbf{c}}^3(q) = \begin{bmatrix} C_{123} p_{CMx3} + a_1 C_1 + a_2 C_{12} \\ S_{123} p_{CMx3} + a_1 S_1 + a_2 S_{12} \\ p_{CMz3} + d_1 - d_2 - d_3 \end{bmatrix} \quad (4.41.c)$$

El centro de masa $\bar{\mathbf{c}}^4(q)$ del cuarto elemento respecto del referencial \mathbf{L}_0 es:

$$\bar{\mathbf{c}}^4(q) = \begin{bmatrix} C_{123} & S_{123} & 0 \\ S_{123} & -C_{123} & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} p_{CMx4} \\ p_{CMy4} \\ p_{CMz4} \end{bmatrix} + \begin{bmatrix} a_1 C_1 + a_2 C_{12} \\ a_1 S_1 + a_2 S_{12} \\ d_1 - d_2 - d_3 - q_4 \end{bmatrix}$$

$$\bar{\mathbf{c}}^4(q) = \begin{bmatrix} C_{123} p_{CMx4} + a_1 C_1 + a_2 C_{12} \\ S_{123} p_{CMx4} + a_1 S_1 + a_2 S_{12} \\ -p_{CMz4} + d_1 - d_2 - d_3 - q_4 \end{bmatrix} \quad (4.41.d)$$

La matriz jacobiana $\mathbf{J}^k(q)$ para el *ROB-DEX* de orden 6×4 expresada con respecto al sistema de coordenadas base \mathbf{L}_0 que se aplica al vector del centro de masa $\mathbf{c}^k(q)$ de cada elemento del robot, se formula utilizando la expresión (4.11) y haciendo uso de las ecuaciones (4.41).

La matriz jacobiana $\mathbf{J}^1(q)$ para el centro de masa del primer enlace es:

$$J^1(q) = \begin{bmatrix} \partial \bar{c}_x^1 / \partial q_1 & 0 & 0 & 0 \\ \partial \bar{c}_y^1 / \partial q_1 & 0 & 0 & 0 \\ \partial \bar{c}_z^1 / \partial q_1 & 0 & 0 & 0 \\ \xi_1 z_x^0 & 0 & 0 & 0 \\ \xi_1 z_y^0 & 0 & 0 & 0 \\ \xi_1 z_z^0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} J_v^1(q) \\ J_w^1(q) \end{bmatrix}$$

$$J^1(q) = \begin{bmatrix} -S_1 p_{CMx1} - a_1 S_1 & 0 & 0 & 0 \\ C_1 p_{CMx1} + a_1 C_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} J_v^1(q) \\ J_w^1(q) \end{bmatrix} \quad (4.42.a)$$

La matriz jacobiana $\mathbf{J}^2(q)$ para el centro de masa del segundo enlace es:

$$J^2(q) = \begin{bmatrix} \partial \bar{c}_x^2 / \partial q_1 & \partial \bar{c}_x^2 / \partial q_2 & 0 & 0 \\ \partial \bar{c}_y^2 / \partial q_1 & \partial \bar{c}_y^2 / \partial q_2 & 0 & 0 \\ \partial \bar{c}_z^2 / \partial q_1 & \partial \bar{c}_z^2 / \partial q_2 & 0 & 0 \\ \xi_1 z_x^0 & \xi_2 z_x^1 & 0 & 0 \\ \xi_1 z_y^0 & \xi_2 z_y^1 & 0 & 0 \\ \xi_1 z_z^0 & \xi_2 z_z^1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} J_v^2(q) \\ J_w^2(q) \end{bmatrix}$$

$$J^2(q) = \begin{bmatrix} -S_{12}p_{CMx2} - a_2S_{12} - a_1S_1 & -S_{12}p_{CMx2} - a_2S_{12} & 0 & 0 \\ C_{12}p_{CMx2} + a_2C_{12} + a_1C_1 & C_{12}p_{CMx2} + a_2C_{12} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} = \begin{matrix} J_v^2(q) \\ \\ \\ J_w^2(q) \end{matrix} \quad (4.42.b)$$

La matriz jacobiana $J^3(q)$ para el centro de masa del tercer enlace es:

$$J^3(q) = \begin{bmatrix} \partial \bar{c}_x^3 / \partial q_1 & \partial \bar{c}_x^3 / \partial q_2 & \partial \bar{c}_x^3 / \partial q_3 & 0 \\ \partial \bar{c}_y^3 / \partial q_1 & \partial \bar{c}_y^3 / \partial q_2 & \partial \bar{c}_y^3 / \partial q_3 & 0 \\ \partial \bar{c}_z^3 / \partial q_1 & \partial \bar{c}_z^3 / \partial q_2 & \partial \bar{c}_z^3 / \partial q_3 & 0 \\ \hline \xi_1 z_x^0 & \xi_2 z_x^1 & \xi_3 z_x^2 & 0 \\ \xi_1 z_y^0 & \xi_2 z_y^1 & \xi_3 z_y^2 & 0 \\ \xi_1 z_z^0 & \xi_2 z_z^1 & \xi_3 z_z^2 & 0 \end{bmatrix} = \begin{matrix} J_v^3(q) \\ \\ \\ J_w^3(q) \end{matrix}$$

$$J^3(q) = \begin{bmatrix} -S_{123}p_{CMx3} - a_2S_{12} - a_1S_1 & -S_{123}p_{CMx3} - a_2S_{12} & -S_{123}p_{CMx3} & 0 \\ C_{123}p_{CMx3} + a_2C_{12} + a_1C_1 & C_{123}p_{CMx3} + a_2C_{12} & C_{123}p_{CMx3} & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix} = \begin{matrix} J_v^3(q) \\ \\ \\ J_w^3(q) \end{matrix} \quad (4.42.c)$$

La matriz jacobiana $J^4(q)$ para el centro de masa del cuarto enlace es:

$$J^4(q) = \begin{bmatrix} \frac{\partial \bar{c}_x^4}{\partial q_1} & \frac{\partial \bar{c}_x^4}{\partial q_2} & \frac{\partial \bar{c}_x^4}{\partial q_3} & \frac{\partial \bar{c}_x^4}{\partial q_4} \\ \frac{\partial \bar{c}_y^4}{\partial q_1} & \frac{\partial \bar{c}_y^4}{\partial q_2} & \frac{\partial \bar{c}_y^4}{\partial q_3} & \frac{\partial \bar{c}_y^4}{\partial q_4} \\ \frac{\partial \bar{c}_z^4}{\partial q_1} & \frac{\partial \bar{c}_z^4}{\partial q_2} & \frac{\partial \bar{c}_z^4}{\partial q_3} & \frac{\partial \bar{c}_z^4}{\partial q_4} \\ \hline \xi_1 z_x^0 & \xi_2 z_x^1 & \xi_3 z_x^2 & \xi_4 z_x^3 \\ \xi_1 z_y^0 & \xi_2 z_y^1 & \xi_3 z_y^2 & \xi_4 z_y^3 \\ \xi_1 z_z^0 & \xi_2 z_z^1 & \xi_3 z_z^2 & \xi_4 z_z^3 \end{bmatrix} = \begin{bmatrix} J_V^4(q) \\ \hline J_W^4(q) \end{bmatrix}$$

$$J^4(q) = \begin{bmatrix} -S_{123} p_{CMx4} - a_2 S_{12} - a_1 S_1 & -S_{123} p_{CMx4} - a_2 S_{12} & -S_{123} p_{CMx4} & 0 \\ C_{123} p_{CMx4} + a_2 C_{12} + a_1 C_1 & C_{123} p_{CMx4} + a_2 C_{12} & C_{123} p_{CMx4} & 0 \\ 0 & 0 & 0 & -1 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix} = \begin{bmatrix} J_V^4(q) \\ \hline J_W^4(q) \end{bmatrix}$$

(4.42.d)

Las masas m_k de cada uno de los elementos del *ROB-DEX* tienen aproximadamente las siguientes magnitudes:

$$\begin{aligned}
m_1 &= 22.4 \text{ Kgms} \\
m_2 &= 12.2 \text{ Kgms.} \\
m_3 &= 4.4 \text{ Kgms.} \\
m_4 &= 3.0 \text{ Kgms.} + \text{ masa del objeto a llevar} \cong 5 \text{ Kgm.}
\end{aligned}$$

43)

Para determinar el tensor de inercia $\mathbf{D}^k(q)$ de cada elemento del *ROB-DEX* debido al movimiento traslacional y rotacional con respecto al referencial base \mathbf{L}_0 se hace uso de la expresión (4.15) combinando las ecuaciones (4.39), (4.41) y (4.42).

El tensor de inercia $\mathbf{D}^1(q)$ para el primer elemento es:

$$\mathbf{D}^1(q) = [J_V^1(q)]^T m_1 J_V^1(q) + [J_W^1(q)]^T D_1 J_W^1(q)$$

$$\mathbf{D}^1(q) = m_1 \begin{bmatrix} d_{11}^1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + I_{zz1} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (4.44.a)$$

donde:

$$d_{11}^1 = (p_{CMx1} + a_1)^2$$

El tensor de inercia $\mathbf{D}^2(q)$ para el segundo elemento es:

$$\mathbf{D}^2(q) = [J_V^2(q)]^T m_2 J_V^2(q) + [J_W^2(q)]^T D_2 J_W^2(q)$$

$$\mathbf{D}^2(q) = m_2 \begin{bmatrix} d_{11}^2 & d_{12}^2 & 0 & 0 \\ d_{21}^2 & d_{22}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + I_{zz2} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (4.44.b)$$

donde:

$$d_{11}^2 = (p_{CMx2} + a_2)^2 + a_1^2 - 2C_2 a_1 (p_{CMx2} + a_2)$$

$$d_{21}^2 = d_{12}^2 = (p_{CMx2} + a_2)^2 - a_1 C_2 (p_{CMx2} + a_2)$$

$$d_{22}^2 = (p_{CMx2} + a_2)^2$$

El tensor de inercia $\mathbf{D}^3(q)$ para el tercer elemento es:

$$\mathbf{D}^3(q) = [J_V^3(q)]^T m_3 J_V^3(q) + [J_W^3(q)]^T D_3 J_W^3(q)$$

$$\mathbf{D}^3(q) = m_3 \begin{bmatrix} d_{11}^3 & d_{12}^3 & d_{13}^3 & 0 \\ d_{21}^3 & d_{22}^3 & d_{23}^3 & 0 \\ d_{31}^3 & d_{32}^3 & d_{33}^3 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + I_{zz3} \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (4.44.c)$$

donde:

$$d_{11}^3 = p_{CMx3}^2 + a_1^2 + a_2^2 + 2a_1a_2C_2 + 2a_1p_{CMx3}C_{23} + 2a_2p_{CMx3}C_3$$

$$d_{21}^3 = d_{12}^3 = p_{CMx3}^2 + a_2^2 + a_1a_2C_2 + a_1p_{CMx3}C_{23} + 2a_2p_{CMx3}C_3$$

$$d_{22}^3 = p_{CMx3}^2 + a_2^2 + 2a_2p_{CMx3}C_3$$

$$d_{31}^3 = d_{13}^3 = p_{CMx3}^2 + a_1p_{CMx3}C_{23} + a_2p_{CMx3}C_3$$

$$d_{32}^3 = d_{23}^3 = p_{CMx3}^2 + a_2p_{CMx3}C_3$$

$$d_{33}^3 = p_{CMx3}^2$$

El tensor de inercia $\mathbf{D}^4(q)$ para el cuarto elemento es:

$$\mathbf{D}^4(q) = [J_V^4(q)]^T m_4 J_V^4(q) + [J_W^4(q)]^T D_4 J_W^4(q)$$

$$\mathbf{D}^4(q) = m_4 \begin{bmatrix} d_{11}^4 & d_{12}^4 & d_{13}^4 & 0 \\ d_{21}^4 & d_{22}^4 & d_{23}^4 & 0 \\ d_{31}^4 & d_{32}^4 & d_{33}^4 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} + I_{zz4} \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (4.44.d)$$

donde:

$$d_{11}^4 = p_{CMx4}^2 + a_1^2 + a_2^2 + 2a_1a_2C_2 + 2a_1p_{CMx4}C_{23} + 2a_2p_{CMx4}C_3$$

$$d_{21}^4 = d_{12}^4 = p_{CMx4}^2 + a_2^2 + a_1a_2C_2 + a_1p_{CMx4}C_{23} + 2a_2p_{CMx4}C_3$$

$$d_{22}^4 = p_{CMx4}^2 + a_2^2 + 2a_2p_{CMx4}C_3$$

$$d_{31}^4 = d_{13}^4 = p_{CMx4}^2 + a_1p_{CMx4}C_{23} + a_2p_{CMx4}C_3$$

$$d_{32}^4 = d_{23}^4 = p_{CMx4}^2 + a_2p_{CMx4}C_3$$

$$d_{33}^4 = p_{CMx4}^2$$

Para determinar el tensor de inercia $\mathbf{D}(q) \in \mathbf{R}^{4 \times 4}$ del *ROB-DEX* se aplica la ecuación (4.16) de la siguiente manera:

$$\mathbf{D}(q) = \sum_{k=1}^4 \mathbf{D}^k(q) = \begin{bmatrix} d_{11} & d_{12} & d_{13} & 0 \\ d_{21} & d_{22} & d_{23} & 0 \\ d_{31} & d_{32} & d_{33} & 0 \\ 0 & 0 & 0 & m_4 \end{bmatrix} \quad (4.45)$$

donde:

$$d_{11} = m_1 d_{11}^1 + m_2 d_{11}^2 + m_3 d_{11}^3 + m_4 d_{11}^4 + I_{zz1} + I_{zz2} + I_{zz3} + I_{zz4}$$

$$d_{12} = d_{21} = m_2 d_{12}^2 + m_3 d_{12}^3 + m_4 d_{12}^4 + I_{zz2} + I_{zz3} + I_{zz4}$$

$$d_{22} = m_2 d_{22}^2 + m_3 d_{22}^3 + m_4 d_{22}^4 + I_{zz2} + I_{zz3} + I_{zz4}$$

$$d_{13} = d_{31} = m_3 d_{13}^3 + m_4 d_{13}^4 + I_{zz3} + I_{zz4}$$

$$d_{23} = d_{32} = m_3 d_{23}^3 + m_4 d_{23}^4 + I_{zz3} + I_{zz4}$$

$$d_{33} = m_3 d_{33}^3 + m_4 d_{33}^4 + I_{zz3} + I_{zz4}$$

La energía cinética para el *ROB-DEX* se determina sustituyendo la ecuación (4.45) en la ecuación (4.17) como sigue:

$$E_c(q, \dot{q}) = \frac{1}{2} \dot{q}^T \mathbf{D}(q) \dot{q} \quad (4.46)$$

donde \dot{q} es el vector velocidad 4×1 de las articulaciones del *ROB-DEX*.

La suma ponderada $\bar{\mathbf{c}}^k(q)$ de los centros de gravedad de los 4 eslabones del *ROB-DEX* por sus respectivas masas, se calcula reemplazando las expresiones (4.41) en la expresión (4.20) obteniéndose lo siguiente:

$$\bar{\mathbf{c}}(q) = \sum_{k=1}^4 m_k \bar{\mathbf{c}}^k(q) = \begin{bmatrix} \bar{c}_x(q) \\ \bar{c}_y(q) \\ \bar{c}_z(q) \end{bmatrix} = \begin{bmatrix} m_1 \bar{c}_x^1 + m_2 \bar{c}_x^2 + m_3 \bar{c}_x^3 + m_4 \bar{c}_x^4 \\ m_1 \bar{c}_y^1 + m_2 \bar{c}_y^2 + m_3 \bar{c}_y^3 + m_4 \bar{c}_y^4 \\ m_1 \bar{c}_z^1 + m_2 \bar{c}_z^2 + m_3 \bar{c}_z^3 + m_4 \bar{c}_z^4 \end{bmatrix} \quad (4.47)$$

La energía potencial del *ROB-DEX* se determina reemplazando la ecuación (4.47) en la ecuación (4.21) de la siguiente manera:

$$E_p(q) = -g^T \bar{\mathbf{c}}(q) = \begin{bmatrix} 0 & 0 & g_0 \end{bmatrix} \begin{bmatrix} \bar{c}_x(q) \\ \bar{c}_y(q) \\ \bar{c}_z(q) \end{bmatrix} = g_0 \bar{c}_z(q) \quad (4.48)$$

Con las expresiones (4.46) y (4.48) se forma la ecuación de *Lagrange* para el *ROB-DEX* y tiene la siguiente forma:

$$L(q, \dot{q}) = \frac{1}{2} \dot{q}^T \mathbf{D}(q) \dot{q} + g^T \bar{\mathbf{c}}(q) \quad (4.49)$$

Los valores de los coeficientes de viscosidad $b(\dot{q})$ para cada uno de los elementos se pueden aproximar de la siguiente manera:

$$\begin{aligned} b_1 &= 80 && \text{Kgm.cm.sec / rad.} \\ b_2 &= 70 && \text{Kgm.cm.sec / rad.} \\ b_3 &= 60 && \text{Kgm.cm.sec / rad.} \\ b_4 &= 100 && \text{Kgm.cm.sec.} \end{aligned} \quad (4.50)$$

La evaluación de la dinámica del *ROB-DEX* se resuelve por computador usando la expresión (4.33) o aplicando directamente el algoritmo *Lagrange-Euler*, para esto se usan las magnitudes instantáneas de articulación, velocidad y aceleración y los valores de los parámetros del robot (masas, centros de masas, momentos de inercias, longitudes de los eslabones, etc.). Las soluciones encontradas se reemplazan en las expresiones (4.31) y (4.32) y con los valores de la fricción viscosa de la ecuación (4.24) resulta finalmente la expresión de la dinámica del *ROB-DEX* por el método de *Lagrange-Euler*.

Desarrollando analíticamente, también se obtiene una expresión dinámica para el *ROB-DEX* en términos funcionalmente explícitos, mediante una expresión vectorial

estructurada, con una interpretación física de sus términos que resulta de realizar una serie de operaciones y se puede encontrar una ecuación para el modelo dinámico de la siguiente manera:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ f_4 \end{bmatrix} = \begin{bmatrix} d_{11} & d_{12} & d_{13} & 0 \\ d_{21} & d_{22} & d_{23} & 0 \\ d_{31} & d_{32} & d_{33} & 0 \\ 0 & 0 & 0 & m_4 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \\ \ddot{d}_4 \end{bmatrix} + \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ h_4 \end{bmatrix} + \begin{bmatrix} b_1 \dot{\theta}_1 \\ b_2 \dot{\theta}_2 \\ b_3 \dot{\theta}_3 \\ b_4 \dot{d}_4 \end{bmatrix} \quad (4.51)$$

La expresión (4.51) se puede expresar desacoplando la cuarta articulación prismática de las tres primeras articulaciones rotacionales de la siguiente manera:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} + \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} + \begin{bmatrix} b_1 \dot{\theta}_1 \\ b_2 \dot{\theta}_2 \\ b_3 \dot{\theta}_3 \end{bmatrix} \quad (4.52.a)$$

$$f_4 = m_4 \ddot{d}_4 + h_4 + b_4 \dot{d}_4$$

$$f_4 = m_4 \ddot{d}_4 - m_4 g_o + b_4 \dot{d}_4 \quad (4.52.b)$$

De las expresiones (4.52) podemos concluir que los torques para accionar a las tres primeras articulaciones dependen del efecto inercial de los eslabones del robot y del acoplamiento de velocidades que existe entre éstas, más no del efecto de la gravedad, mientras que la fuerza desarrollada por el cuarto motor depende de la masa del cuarto elemento y del efecto gravitatorio mas no del efecto de acoplo de velocidades. Las fuerzas de fricción intervienen en las cuatro articulaciones.

La expresión (4.51) también se puede expresar en forma simplificada dada por la ecuación (4.36) como:

$$\tau(t) = D(q(t)) \ddot{q}(t) + c(q(t), \dot{q}(t)) + h(q(t)) + b(\dot{q}(t)) \quad (4.53)$$

4.5 Propiedades del modelo dinámico

Ya sea en la etapa de diseño mecánico o en la de análisis del comportamiento dinámico de un robot, no basta con conocer el modelo dinámico. El conocimiento de su estructura y propiedades de dicho modelo hará más eficiente al trabajo, así como la síntesis de los controladores. Por tal motivo, se hará un breve análisis de las propiedades del modelo obtenido.

- **Propiedad de la matriz de inercia**

La matriz de inercia $D(q)$ fue obtenida a partir del tensor de inercia, por lo cual siempre resulta ser simétrica y definida positiva:

$$D(q) = D^T(q) \quad (4.54)$$

$$x^T D(q) > 0 ; \quad \forall x \in \mathbf{R}^n \quad (4.55)$$

Otra propiedad importante de la matriz de inercia es que está acotada tanto por arriba como por abajo. Esto es:

$$\mu_1 \mathbf{I} \leq D(q) \leq \mu_2 \mathbf{I} \quad (4.56)$$

Los escalares μ_1 y μ_2 se pueden calcular fácilmente para un determinado robot. Del mismo modo, la inversa de la matriz de inercia también está acotada:

$$\frac{1}{\mu_1} \mathbf{I} \leq D^{-1}(q) \leq \frac{1}{\mu_2} \mathbf{I} \quad (4.57)$$

Cuando el robot tiene solo articulaciones rotacionales, μ_1 y μ_2 son constantes pues los términos $d_{ij}(q)$ incluyen las variables articulares, a través de funciones trascendentales (seno y coseno), las que están acotadas por la unidad. Cuando se tiene articulaciones translacionales, éstas cotas son funciones escalares conocidas de q . Otra manera de expresar el acotamiento de la matriz de inercia es la siguiente:

$$d_1 \leq \|D(q)\| \leq d_2 \quad (4.58)$$

donde: los escalares d_1 y d_2 se pueden definir con cualquier norma inducida.

- **Propiedad del vector de acoplamiento en velocidad**

El término acoplamiento en velocidad dado por la ecuación (4.35) puede expandirse de la siguiente manera:

$$c_i(q, \dot{q}) = \sum_{k=1}^n C_{kk}^i(q) \dot{q}_k^2 + \sum_{k=1}^n \sum_{j \neq k}^n C_{kj}^i(q) \dot{q}_k \dot{q}_j \quad (4.59)$$

Estos dos términos son generados respectivamente por la diagonal y por los términos fuera de la diagonal de las matrices de acoplamiento en velocidad. El primero de ellos representa las fuerzas centrífugas $C_{kk}^i(q) \dot{q}_k^2$ correspondiente a la fuerza centrífuga actuando sobre la i -ésima articulación originada por el movimiento de la k -ésima articulación. El segundo término representa la fuerza de Coriolis $C_{kj}^i(q) \dot{q}_k \dot{q}_j$ en la i -ésima articulación producida por el movimiento combinado de las articulaciones k y j con $k \neq j$.

Debido a su naturaleza, el vector de acoplamiento en velocidad está acotado por arriba por una función cuadratura de las velocidades, de modo que:

$$\|c(q, \dot{q})\| \leq \mu_c \|\dot{q}\|^2 \quad (4.60)$$

En donde $\|\cdot\|$ es una norma apropiada.

Para todo robot con articulaciones rotacionales, el escalar μ_c es una constante independiente de q por las razones expuestas en el inciso anterior. Cuando se tienen articulaciones prismáticas, μ_c es una función escalar conocida de q . Para cualquier robot siempre existe una factorización no única, del vector de acoplamiento en

velocidad, tal que:

$$c(q, \dot{q}) = C_s(q, \dot{q}) \dot{q} \quad (4.61)$$

$$S(q, \dot{q}) \Delta C_s(q) - 2C_s(q, \dot{q}) \quad (4.62)$$

en donde $C_s(q, \dot{q})$ es una matriz anti-simétrica de modo que $x^T Sx = 0$ para toda $x \in \mathbf{R}^n$.

- **Propiedad del término gravitatorio**

El término gravitatorio $h(q)$ dado por la ecuación (4.31) también está acotado por arriba para cualquier robot con articulaciones rotacionales por una constante μ_g , de la siguiente manera:

$$\|h(q)\| \leq \mu_g \quad (4.63)$$

En donde $\|\cdot\|$ es cualquier norma vectorial apropiada. Cuando alguna de las articulaciones del robot es prismática, la cota μ_g es una función escalar del vector de coordenadas articulares q , la cual puede determinarse fácilmente para un robot dado.

- **Linealidad de los parámetros**

Una propiedad muy importante del modelo dinámico expresado en la ecuación (4.36), de mucha utilidad en la síntesis de controladores adaptables, es su linealidad en los parámetros. Esto es, a pesar de que cualquier robot está representado por una función no lineal desde el torque aplicado hasta las variables articulares producidas, dicho modelo dinámico admite una factorización lineal en sus parámetros dinámicos.

Esto se puede expresar de la siguiente manera:

$$\begin{aligned} \tau(t) &= D(q(t)) \ddot{q}(t) + c(q(t), \dot{q}(t)) + h(q(t)) + b(\dot{q}(t)) \\ \tau(t) &= W(q(t), \dot{q}(t), \ddot{q}(t)) \Phi \end{aligned} \quad (4.64)$$

donde Φ es el vector de parámetros dinámicos del robot, mientras que $W(q, \dot{q}, \ddot{q})$ es una matriz de mediciones formada por funciones conocidas dependientes de posiciones, velocidades y aceleraciones articulares.

A menudo los parámetros dinámicos son complejos de medir por la variedad de materiales y de las formas con que se construyen sus eslabones; o bien son cambiantes, lo cual complica su medición. El interés de esta propiedad radica en que proporciona un modelo para identificación y/o estimación de dichos parámetros, útil también para el diseño de controladores adaptables.

- **Pasividad y conservación de la energía**

La pasividad es una propiedad que permite analizar la estabilidad de sistemas no lineales en base a mediciones de entrada y salida. Sea un sistema cuyo vector de entrada $u(t)$ tiene la misma dimensión que su vector de salida $y(t)$. Se dice que este sistema es pasivo si para todo $T > 0$ finito y algún $\gamma > -\infty$ y se tiene que:

$$\int_0^T y(t)^T u(t) dt \geq \gamma \quad (4.65)$$

Un sistema pasivo no genera energía y desde el punto de vista de control no es inestable. El modelo representado por la ecuación (4.36) es pasivo de $\tau(t)$ a $\dot{q}(t)$, sin considerar el término de fricción $b(\dot{q})$, éste no es muy restrictivo pues la fricción es un fenómeno disipativo, independientemente del modelo considerado, de modo que la función $b(\dot{q})$ yace en el primer y tercer cuadrante (Fig. C.1), de modo que:

$$\dot{q}^T b(\dot{q}) \geq 0 \quad (4.66)$$

La naturaleza disipativa de la fricción permite conservar la propiedad de pasividad del modelo dinámico del robot. Además, permite incrementar el ancho de banda del sistema.

- **Representación de estado**

Gracias a la propiedad establecida en la ecuación (4.55), es posible tener una representación de estado del modelo dinámico del robot expresado en la ecuación (4.36). Esto es, definiendo el siguiente vector de estado:

$$x = [q^T \quad \dot{q}^T]^T \in \mathbf{R}^{2n} \quad (4.67)$$

Se puede obtener la siguiente representación en el espacio de estado para la ecuación que rige el comportamiento dinámico de un robot:

$$\dot{x} = \begin{bmatrix} \dot{q} \\ -D^{-1}(q)(c(q, \dot{q}) + h(q) + b(\dot{q})) \end{bmatrix} + \begin{bmatrix} 0 \\ D^{-1}(q) \end{bmatrix} \tau \quad (4.68)$$

la cual es una representación de estado no lineal de la forma:

$$\dot{x} = f(q, \dot{q}) + B(q) u \quad (4.69)$$

Con $u = \tau$, y $f(\cdot)$, $B(\cdot)$ funciones no lineales de estado. También es posible obtener una ecuación de estado lineal de la siguiente manera:

$$\dot{x} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ I \end{bmatrix} u \quad (4.70)$$

en donde la entrada de control está definida como sigue:

$$u = -D^{-1}(q)(c(q, \dot{q}) + h(q) + b(\dot{q})) + D^{-1}(q)\tau \quad (4.71)$$

CAPITULO V

SISTEMA DE CONTROL DEL ROBOT

5.1 Introducción

Una vez que la trayectoria nominal del robot es determinada, queda el problema de transmitir los comandos a los actuadores de articulación que harán que el manipulador siga fielmente la trayectoria planificada. Esto se llama el *Problema de control del robot* y numerosas técnicas han sido propuestas para su solución. Una manera es convertir la trayectoria de la configuración de la herramienta $x(t)$ en espacio cartesiano a su correspondiente trayectoria de espacio de articulación $q(t)$ a través de las ecuaciones de la cinemática inversa, y entonces derivar o diferenciar $q(t)$ para determinar las velocidades $\dot{q}(t)$, a las cuales las articulaciones individuales deberán conducirse. Una variación de esta técnica, por ejemplo, es el método de *rate-control* (relación de control) de movimiento resuelto, el cual usa la inversa generalizada de la matriz jacobiana de la configuración de la herramienta. Las formas de control asumen implícitamente que el *hardware* está disponible para regular la velocidad de cada articulación.

Otra alternativa para controlar un brazo robótico consiste en regular el torque τ de cada articulación. Las técnicas de control basadas en el torque se forman directamente sobre el modelo dinámico del robot. Las técnicas de sistemas de retroalimentación lineal son usadas para analizar el control PID de *articulación*

simple o de un solo eje, cuyo método de diseño es ampliamente usado en los actuales robots comerciales. Otros métodos no lineales más sofisticados incluyen un control PD con compensación de la gravedad, la técnica de control de *torque calculado*, el control de estructura variable, control de impedancia, etc.

5.2 El problema de control

El control de un robot manipulador como un problema de seguimiento de trayectoria es una tarea compleja debido al modelo dinámico, no lineal, diferenciable continuamente en el tiempo y multivariable. Este modelo es un sistema simultáneo de n ecuaciones diferenciales de segundo orden ($n =$ grados de libertad del robot).

Un control de un alto desempeño de los robots manipuladores mediante la regulación del torque es una tarea difícil; tal dificultad proviene de la complejidad del modelo dinámico del brazo. Recalcando que las ecuaciones del movimiento de un brazo robótico en general son de la siguiente forma:

$$D(q) \ddot{q} + c(q, \dot{q}) + h(q) + b(\dot{q}) = \tau \quad (5.1)$$

entonces, un robot de n articulaciones es modelado como un sistema simultáneo de n ecuaciones diferenciales de segundo orden no lineales y altamente acopladas. Lo primero es observar que en muchas aplicaciones el robot no tiene que conducirse a alta velocidad. Si \dot{q} representa el vector de la velocidad instantánea de articulación, entonces la medida aproximada de la velocidad total del manipulador es $\|\dot{q}\|$.

Uno de los términos más complejos en el modelo dinámico de un robot es el término de acoplamiento de las velocidades $c(q, \dot{q})$, el cual es generado por las fuerzas centrífugas y de Coriolis. Recalcando de la ecuación (4.36) que la k -ésima componente del vector de acoplamiento de velocidades es de la forma:

$$c_k(q, \dot{q}) = \dot{q}^T C^k(q) \dot{q}$$

en consecuencia, para cada punto q en el espacio de articulación $c(q, \dot{q}) \rightarrow 0$ cuando $\|\dot{q}\| \rightarrow 0$. Por tanto, si la velocidad del robot se reduce suficientemente, el modelo dinámico se simplifica en el sentido que los efectos de las fuerzas centrífugas y de Coriolis se hacen pequeñas en comparación con los términos que no dependen de la velocidad \dot{q} . Si el robot está operando de modo que las articulaciones son activadas una en cada tiempo en vez de hacerlo simultáneamente, entonces la velocidad de acoplamiento debido a la fuerza de Coriolis (pero no las fuerzas centrífugas) desaparecen completamente, independiente de la velocidad del robot.

El término de la fricción $b(\dot{q})$ también depende de la velocidad de articulación; de la ecuación (4.27) vemos que $b(0) = 0$ desde que $\text{sgn}(0) = 0$. Sin embargo, el torque debido a la fricción no se puede hacer arbitrariamente pequeño debido a que $b(\dot{q})$ presenta una discontinuidad en $\dot{q} = 0$. Desde luego, esto es claro de la Fig. (C.1), que los efectos relativos de la fricción estática vienen a ser más pronunciadas cuando la velocidad del brazo es reducida.

Otra característica de muchos robots comerciales que hace algo más fácil su control es la reducción de su velocidad por medio de engranajes entre los actuadores y los eslabones. Si N_k denota la relación de reducción de velocidad desde el eje del actuador al eje de carga de la articulación k , entonces la componente variable de la inercia del brazo, referido al actuador para la articulación k , es reducido por el factor de N_k^2 . Similarmente, el torque debido a la velocidad de acoplamiento $c_k(q, \dot{q})$ y al torque de carga debido a la gravedad $h_k(q)$, son reducidos por un factor de N_k cuando son expresadas respecto al eje del actuador de articulación.

Algunos tipos de robots se diseñan para ser estructuralmente bastante "rígidos" en comparación con uno con sistema de control altamente dinámico. Esto hace al robot más fácil de controlar, pero reduce su velocidad operativa, su potencialidad y su productividad.

La mayoría de los robots comerciales actuales tienen controladores relativamente simples, usando reducción de engranajes y operan a velocidades moderadas. Las tendencias actuales en el diseño de robots están orientados a usar actuadores de alto torque ubicados directamente en las articulaciones del robot. La eliminación de engranajes y otros mecanismos de transmisión de potencia reducen la fricción y el juego entre engranajes, permitiendo además a las articulaciones del robot moverse considerablemente a velocidades altas. El buen desempeño dinámico desarrollado de manipuladores es necesario si los robots industriales van a operar sobre su máxima potencialidad con el fin de incrementar la productividad y mejorar la calidad de la producción. Por esta razón, debemos encaminarnos en un control sofisticado de robots manipuladores que usen modelos dinámicos realistas, aunque ésta sea una tarea compleja.

El desarrollo de algoritmos de control para robots manipuladores actualmente es una tarea de investigación. Este capítulo no incluye un estudio completo de los actuales métodos de control para robots porque muchos de los algoritmos propuestos en la literatura de Robótica son muy complejos y sofisticados, los cuales están aún en desarrollo. Aquí se hace una prueba de los algoritmos de control con una predisposición hacia las técnicas actuales más simples, dado que otras técnicas están fuera del alcance de este trabajo.

5.3 Ecuaciones de estado

El modelo dinámico del brazo de un robot está formulado en la ecuación (5.1) como un sistema de n ecuaciones diferenciales de segundo orden. Para facilitar el control de un robot es útil reformular las ecuaciones de movimiento como un sistema de $2n$ ecuaciones de primer orden llamadas *ecuaciones de estado*. Al hacer esta transformación a la forma de espacio de estado se aísla el vector de aceleración \ddot{q} en la ecuación (5.1). Esto puede hacerse fácilmente debido que la matriz del tensor de inercia del manipulador $D(q)$ es simétrica, definida positiva y siempre no singular. El modelo de espacio de estado del brazo de un robot incluye las expresiones del estado dinámico y la ecuación de salida que viene a ser la cinemática del robot.

Si definimos a la velocidad como $v \triangleq \dot{q}$ y representamos al vector de variables de estado como $x^T \triangleq [q^T, v^T]$, entonces las ecuaciones de movimiento del brazo de un robot en la ecuación (5.1) se puede representar mediante el siguiente modelo de espacio de estado como ecuaciones diferenciales de primer orden:

$$\dot{q} = v \quad (5.2.a)$$

$$\begin{aligned} \dot{v} &= \ddot{q} \\ \dot{v} &= \mathbf{D}^{-1}(q) [\tau - h(q) - c(q, v) - b(v)] \end{aligned} \quad (5.2.b)$$

Las ecuaciones de estado en las ecuaciones (5.2) se puede simplificar escribiendo una ecuación f no lineal de la siguiente manera:

$$\dot{x}(t) = f(q(t), v(t), \tau(t)) \quad (5.3)$$

Si $g(q)$ representa la función de la configuración de la herramienta (posición y orientación), entonces la ecuación de salida $y(t)$ viene a ser:

$$y(t) = g(q(t)) \quad (5.4)$$

El vector $x(t) \in \mathbf{R}^{2n}$ representa el estado del brazo de robot en el tiempo t . El vector de salida $y(t) \in \mathbf{R}^m$ representa la posición y la orientación del efector final y muestra que la configuración de la herramienta del robot depende sólo de las n primeras componentes del vector de estado $x(t)$. El diagrama de bloques del modelo de espacio de estado del brazo del robot se muestra en la Fig. (5.1).

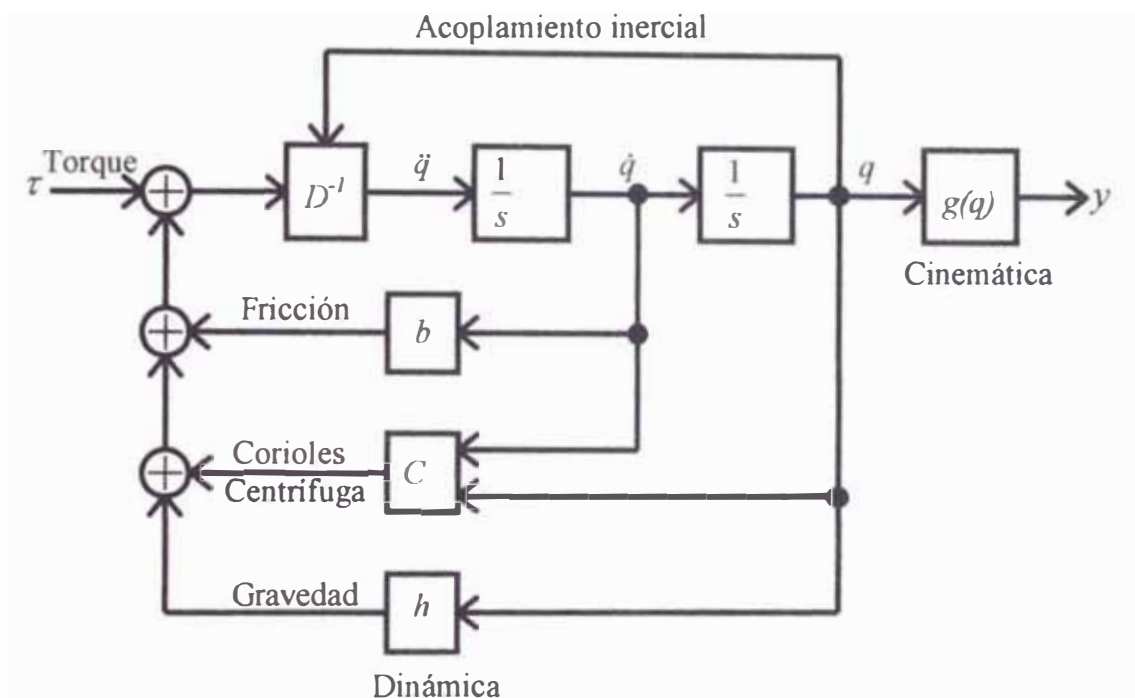


Fig. 5.1 Modelo de espacio de estado de un robot.

Se puede resumir diciendo entonces que el brazo de un robot con n articulaciones se puede modelar por un sistema diferencial de primer orden no lineal de dimensión $2n$, con n entradas que consisten en torques o fuerzas proporcionadas por los actuadores, y m salidas que especifican la configuración del efector final.

5.4 Métodos de control

El control de un robot basado en computador tiene como objetivo mantener la respuesta dinámica del manipulador utilizando algún criterio, método o estrategia de

funcionamiento preespecificado, utilizando el modelo dinámico del manipulador y recibiendo la información especificada como referencia y la información de la retroalimentación sensorial del desplazamiento, velocidad y fuerza/torque de sus articulaciones a lo largo de una trayectoria planificada, usando la cinemática directa y/o inversa. En la Fig. (5.2) se muestra un esquema de diagrama de bloques del control de un robot.

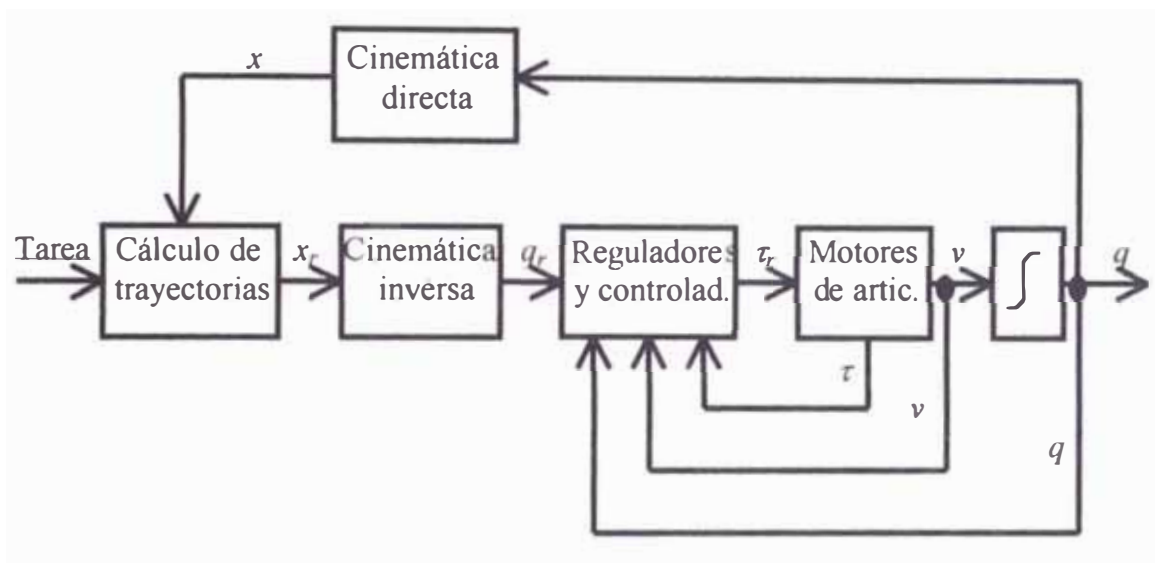


Fig. 5.2 Sistema general de control de un robot.

El sistema de control de un robot permite gobernar en tiempo real los movimientos del robot mediante la información acerca de las magnitudes que se usan como referencia, así como de la información que suministran los sensores de estado interno, disponiendo para ello, de complejos algoritmos de control. Dado que el sistema de control intercambia información con el exterior, existe un bloque de entradas/salidas (E/S) de intercomunicación de alta velocidad para controlar los movimientos del robot necesarios para llevar a cabo una tarea concreta, que puede consistir en movimientos aislados de su estructura o en un programa de movimientos almacenados en la memoria del computador central. Para esto es necesario que el

robot disponga de un sistema de control capaz de asegurar un posicionamiento rápido y preciso del efector final en cualquiera de las condiciones operativas en que pueda utilizarse.

El rendimiento dinámico de un manipulador depende directamente de la eficacia de sus algoritmos de control y de la precisión del modelo dinámico. El control de un robot consiste en obtener modelos dinámicos del brazo del robot físico y luego especificar leyes o estrategias de control correspondientes para conseguir la respuesta y el rendimiento deseado del sistema.

El control de un robot independientemente de sus actuadores se puede clasificar en tres grandes categorías que son

- Controles de movimiento de articulación.
- Controles con movimiento resuelto (en el espacio cartesiano).
- Controles Adaptivos.

Aquí sólo veremos algunos de los métodos de control referente al movimiento de articulación y son: Control posicional PID de simple articulación, control PD con compensación de la gravedad y control mediante el método del torque calculado.

5.4.1 Control posicional PID de simple articulación

Un método común de control de muchos robots comerciales es el controlador PID de articulación simple. Para describir matemáticamente este controlador denotemos $r(t) \in \mathbf{R}^n$ como la *entrada de referencia* deseada que nos gustaría que sigan las variables articulares y denotemos $e(t)$ al *error acarreado* en el tiempo t . Esto es:

$$e(t) \triangleq r(t) - q(t) \quad (5.5)$$

Idealmente, $e(t)$ debe ser idénticamente a cero, pero en la práctica varía de cero, particularmente cuando la entrada de referencia $r(t)$ cambia rápidamente. Una técnica común para controlar un manipulador es emplear n controladores independientes, uno por cada articulación, con la siguiente ley de control:

$$\tau(t) = K_p e(t) + K_I \int_0^t e(\tau) d\tau + K_D \frac{d}{dt} e(t) \quad (5.6)$$

Aquí $\{K_p \ K_I \ K_D\}$ son matrices diagonales de $n \times n$, lo que significa que cada eje está controlado *separadamente*. Hay que notar que la señal $\tau(t)$ del control PID consiste de los términos: proporcional (P) al error, integral (I) del error y derivada (D) del error, de allí su nombre. Las matrices diagonales $\{K_p \ K_I \ K_D\}$ viene a ser parámetros diseñados llamados ganancia *proporcional*, ganancia *integral* y ganancia *derivativa*, respectivamente.

La Fig. (5.3) ilustra el diagrama de bloques que representa a un controlador PID de simple articulación en términos de la transformada de *Laplace*. Cuando $n=1$, las matrices diagonales de ganancia se reducen a matrices de 1×1 con ganancias escalares: $\{k_p \ k_I \ k_D\}$, como parámetros del controlador.

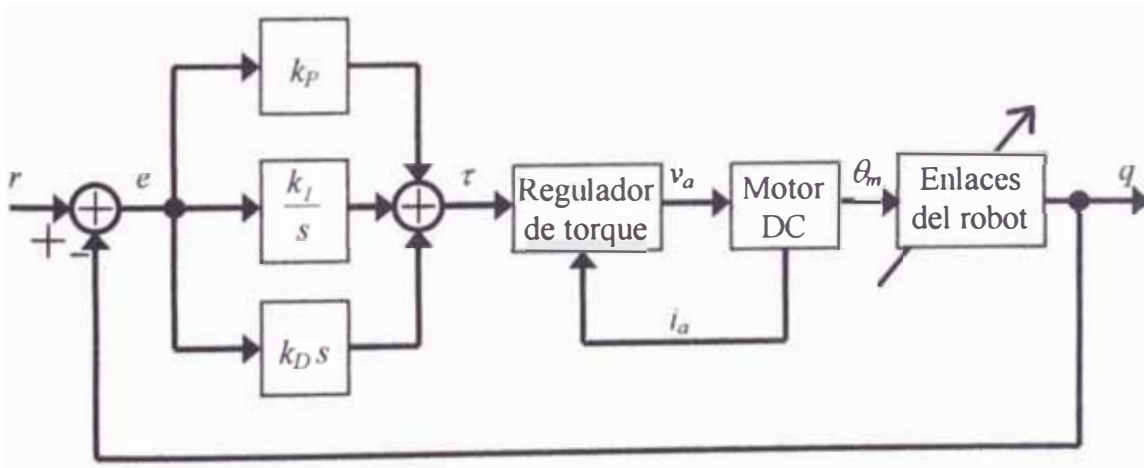


Fig. 5.3 Controlador PID de simple articulación.

La expresión para $\tau(t)$ en la ecuación (5.6) es la que produce el torque de referencia τ que sirve como entrada al subsistema del regulador del torque que controla el torque del motor, sensando la corriente de armadura i_a y variando el voltaje de armadura v_a del motor. La carga para el k -ésimo controlador PID incluye tanto al eslabón k como todos los eslabones restantes hacia el extremo final del brazo. Por tanto, la carga en cada articulación cambia rápidamente durante la operación normal del robot. Esta variación simbólicamente se indica en la Fig. (5.3) con una flecha diagonal a través del bloque de los enlaces del brazo.

- **Modelamiento matemático de un motor DC**

Para analizar el control PID de simple articulación de un robot se modela los subsistemas de la Fig. (5.3) en mayor detalle. Primero consideremos el motor DC incluyendo la carga. Un diagrama esquemático de un servomotor DC de imán permanente controlado por voltaje de armadura con carga mecánica es mostrado en la Fig. (5.4).

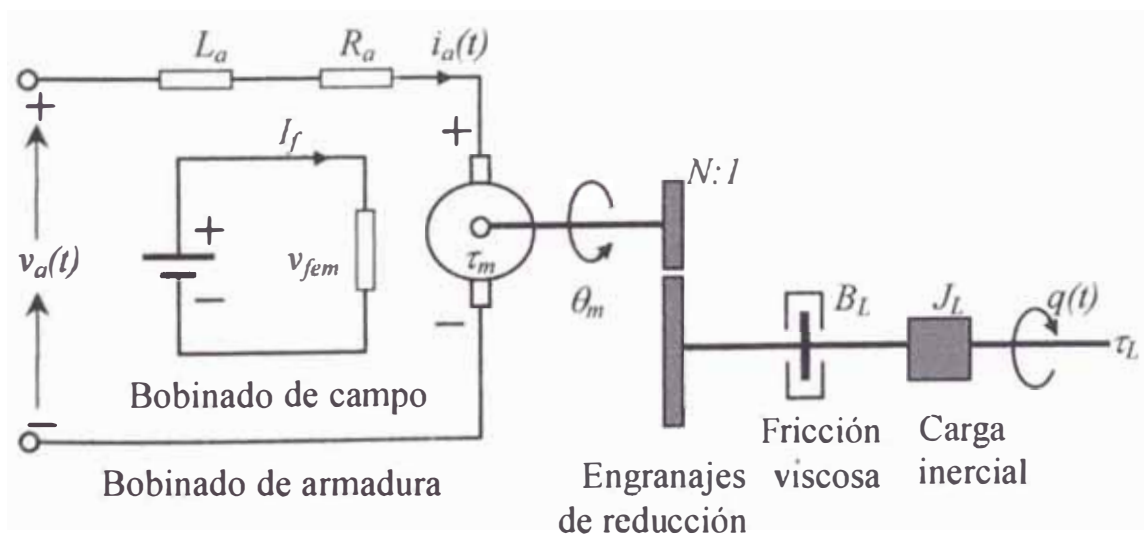


Fig. 5.4 Motor DC controlado por voltaje de armadura.

Para desarrollar el modelo dinámico del motor empezamos a modelar el circuito eléctrico del lado izquierdo de la Fig. (5.4). Aplicando las leyes de voltaje de *Kirchhoff* en la parte eléctrica del bobinado de la armadura se tiene:

$$v_a(t) = L_a \frac{d}{dt} i_a(t) + R_a i_a(t) + v_{fem}(t) \quad (5.7)$$

Aquí la entrada $v_a(t)$ es el voltaje aplicado al bobinado de armadura, mientras que $i_a(t)$ es la corriente de armadura. Los parámetros R_a y L_a son la resistencia e inductancia del bobinado de armadura respectivamente. Finalmente, $v_{fem}(t)$ es el llamado voltaje de fuerza contraelectromotriz, el cual es un voltaje interno que contrarresta a $v_a(t)$ y se incrementa proporcionalmente a medida que aumenta la velocidad en el eje del motor y se expresa como sigue:

$$v_{fem}(t) = K_b \frac{d}{dt} \theta_m(t) \quad (5.8)$$

Aquí $\theta_m(t)$ es el desplazamiento angular en el eje del motor. La constante $K_b > 0$ se denomina constante de voltaje de fuerza contraelectromotriz.

Seguidamente examinamos la parte mecánica del sistema de la Fig. (5.4). Aplicando la segunda ley de *Newton*, la ecuación de movimiento para la masa rotante en el eje del motor es:

$$\tau_m(t) = J \frac{d^2}{dt^2} \theta_m(t) + B \frac{d}{dt} \theta_m(t) \quad (5.9)$$

Aquí $\tau_m(t)$ denota el torque desarrollado en el eje del motor. Los parámetros J y B son *el momento de inercia efectivo* y *el coeficiente de fricción viscosa efectivo* respectivamente, respecto al eje del motor. Si J_m y B_m representan el momento de inercia y el coeficiente de fricción viscosa para el motor y J_L y B_L denotan cantidades

correspondientes para la carga, entonces estas magnitudes se las puede relacionar como:

$$J = J_m + N^2 J_L \quad (5.10.a)$$

$$B = B_m + N^2 B_L \quad (5.10.b)$$

$$q = N\theta_m \quad (5.10.c)$$

Aquí N representa la relación de reducción de engranajes entre el eje del motor (alta velocidad) el eje de la carga (baja velocidad).

Los subsistemas eléctricos y mecánicos están acoplados uno del otro mediante una ecuación algebraica del torque. En general, el torque desarrollado en el eje del motor es proporcional al producto de dos corrientes, la corriente de armadura i_a y la corriente de campo I_f , sin embargo, en un motor DC controlado por voltaje de armadura, I_f es constante; por tanto el torque desarrollado en el eje del motor es:

$$\tau_m(t) = K_a i_a(t) \quad (5.11)$$

La constante $K_a > 0$ es la llamada *constante de torque* del motor.

Para desarrollar una expresión de la función de transferencia del motor DC se toma la transformada de *Laplace* a las ecuaciones (5.7), (5.8) y (5.11); asumiendo las condiciones iniciales nulas y combinando los resultados, se obtiene la ecuación:

$$V_a(s) = (L_a s + R_a) \Gamma_m(s) / K_a + s K_b \Theta_m(s) \quad (5.12)$$

Tomando la transformada de *Laplace* a ambos miembros de la ecuación (5.9), asumiendo que la condiciones iniciales son nulas, para la parte mecánica del sistema de la Fig. (5.4), resulta la siguiente ecuación:

$$\Gamma_m(s) = (J s^2 + B s) \Theta_m(s) \quad (5.13)$$

Desde que el motor DC con carga es comandado por el sistema del regulador de

torque en la Fig. (5.3), es útil desarrollar una *función de transferencia de torque* $G_L(s) \triangleq Q(s)/\Gamma_L(s)$, la cual relaciona la posición angular en el eje de la carga con el torque desarrollado en el eje de carga con respecto al eje del motor. Asumiendo que no hay juego ni deformación elástica en los engranajes, entonces el trabajo realizado en el eje de la carga es igual al trabajo realizado en el eje del motor: $\tau_L q = \tau_m \theta_m$. El torque en el eje de carga se puede escribir usando la ecuación (5.10.c), como:

$$\tau_L(t) = \tau_m(t)/N \quad (5.14)$$

Si el torque deseado en el eje de la carga es visto como una señal de entrada, entonces de las ecuaciones (5.10.c), (5.13) y (5.14) vemos que la función de transferencia del torque τ_L al desplazamiento angular q es como sigue:

$$G_L(s) = \frac{Q(s)}{\Gamma_L(s)} = \frac{N^2}{s(Js + B)} \quad (5.15)$$

Así, la función de transferencia que relaciona la entrada $\Gamma_L(s)$ con la salida $Q(s)$ es un sistema de segundo orden con un polo en el origen. De la ecuación (5.15) se deduce que la *constante de tiempo* T_m de la función de transferencia del torque es:

$$T_m = J/B$$

Si se ignoran los efectos de fricción viscosa, es decir $B=0$, la función de transferencia del torque se reduce puramente a la inercia como:

$$G_L(s) = N^2 / (Js^2) \quad (5.16)$$

Si el voltaje de armadura v_a se toma como la entrada al motor, entonces usando las ecuaciones (5.10.c), (5.12) y (5.13) se puede dibujar el esquema de diagrama de bloques para el motor tal como se muestra en la Fig. (5.5).

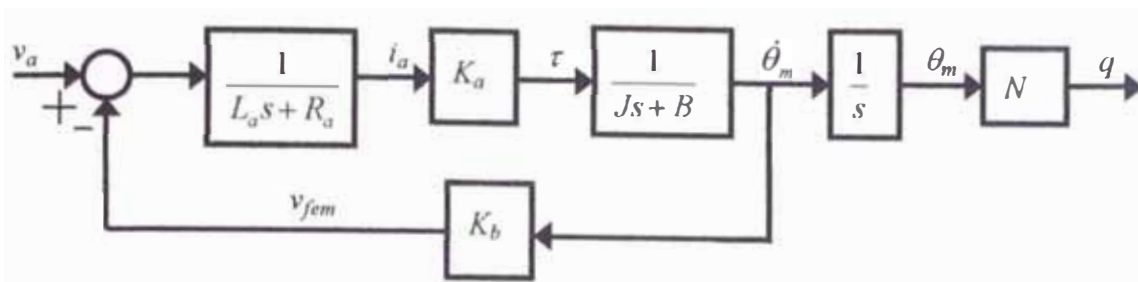


Fig. 5.5 Esquema de un motor DC controlado por voltaje de armadura.

La expresión para la función de transferencia $G_V(s) \triangleq Q(s)/V_a(s)$ que relaciona la posición angular de la carga q con el voltaje de armadura v_a se puede escribir en base a la Fig. (5.5) como sigue:

$$G_V(s) = \frac{Q(s)}{V_a(s)} = \frac{N K_a}{s [(J s + B)(L_a s + R_a) + K_a K_b]} \quad (5.17)$$

Si se desprecia L_a en la expresión (5.17) es decir $L_a \approx 0$, entonces la función de transferencia $G_V(s)$ se convierte en:

$$G_V(s) = \frac{N K_a}{s [J R_a s + (B R_a + K_a K_b)]}; \quad L_a \approx 0 \quad (5.18)$$

La función de transferencia de la ecuación (5.18) se puede expresar en una forma general como:

$$G_V(s) = \frac{N K_m}{s (T_m s + 1)} \quad (5.19)$$

donde los parámetros K_m y T_m de la ecuación (5.19) se llaman ganancia del motor y constante de tiempo del motor respectivamente y denotan como sigue:

$$K_m = \frac{K_a}{B R_a + K_a K_b}, \quad T_m = \frac{J R_a}{B R_a + K_a K_b}$$

- **Regulador de torque**

Considerando el subsistema del *regulador de torque* en la Fig. (5.3); el torque de un motor DC puede ser controlado indirectamente al variar el voltaje de armadura v_a aplicado. Como el torque τ es proporcional a la corriente de armadura i_a , para regular el torque (o corriente de armadura), insertamos un pequeño resistor R_s como *sensor de corriente* en serie con la bobina de armadura a fin de generar un voltaje de retroalimentación v_s , tal como se muestra en la Fig. (5.6).

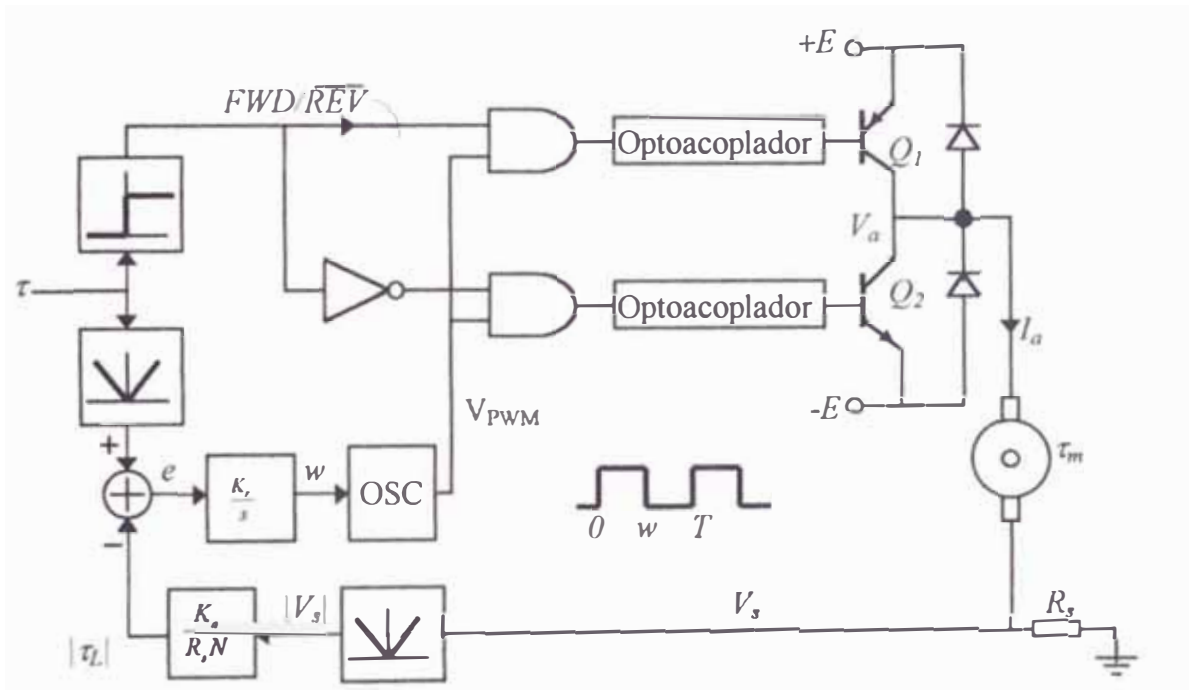


Fig. 5.6 Sistema del regulador de torque.

El elemento clave de control en el sistema de retroalimentación en la Fig. (5.6) es el oscilador OSC . La salida del oscilador V_{PWM} , es un tren de pulsos con *ancho de pulso* w y *periodo* T , donde T debe ser pequeño en comparación con la constante de tiempo del motor T_m . Así V_{PWM} es una *señal modulada por ancho de pulso* (PWM), tal como se muestra en la Fig. (5.7).

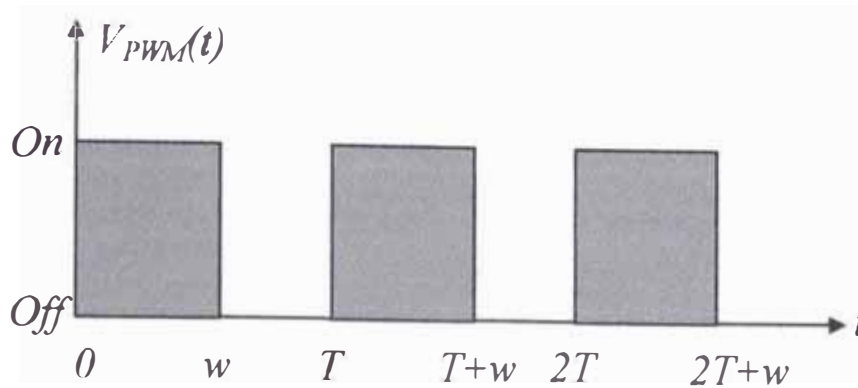


Fig. 5.7 Señal de control por modulación por ancho de pulso.

La señal V_{PWM} conmuta en corte y saturación a los transistores de potencia Q_1 y Q_2 de la Fig. (5.6) mediante optoacopladores. Dependiendo del signo de τ , uno de los transistores no conducirá mientras que el otro conduce en forma pulsante "on-off" por V_{PWM} . Así, el signo de τ controla la dirección en la que el eje del motor debe girar; mientras que el ancho de pulso w controla el torque desarrollado en el eje del motor. El torque se controla variando el valor promedio del voltaje de armadura $v_a(t)$.

Dado un voltaje de alimentación DC de magnitud E , el *valor promedio* de la magnitud del voltaje de armadura es $V_a = wE/T$, donde $0 < w < T$. Llevando a la transformada de *Laplace* asumiendo condiciones iniciales cero se obtiene:

$$\bar{V}_a = \frac{W(s)E}{T} \quad (5.20)$$

El voltaje de armadura promedio, o componente directa DC del voltaje de armadura $v_a(t)$ es muy importante porque un motor de corriente continua DC es esencialmente un filtro pasa bajo que elimina tanto al armónico principal como a los armónicos más altos de la señal periódica $v_a(t)$, tanto como el período T del generador *PWM* sea pequeño en comparación con la constante de tiempo del motor T_m , es decir, se debe cumplir $T \ll T_m$.

Notar en la Fig. (5.6) que el regulador de torque emplea una integral en el control retroalimentado con una ganancia de K_r . Para desarrollar una función de transferencia de lazo cerrado para el regulador de torque analizamos la señal $W(s)$. Referente a la Fig. (5.6) y usando las ecuaciones (5.11) y (5.14), obtenemos una expresión de $W(s)$ de la siguiente manera:

$$\begin{aligned}
 W(s) &= \frac{K_r}{s} E(s) \\
 &= \frac{K_r}{s} \left\{ |\Gamma(s)| - K_a |V_s(s)| / (NR_s) \right\} \\
 &= \frac{K_r}{s} \left\{ |\Gamma(s)| - |K_a V_s(s) / (NR_s)| \right\} \\
 &= \frac{K_r}{s} \left\{ |\Gamma(s)| - |K_a I_a(s) / N| \right\} \\
 &= \frac{K_r}{s} \left\{ |\Gamma(s)| - |\Gamma_m(s) / N| \right\} \\
 &= \frac{K_r}{s} \left\{ |\Gamma(s)| - |\Gamma_L(s)| \right\} \tag{5.21}
 \end{aligned}$$

Sustituyendo la ecuación (5.21) en la ecuación (5.20), reemplazando $\bar{V}_a(s)$ con $V_a(s)$, los signos de la magnitud se pueden dejar de lado porque el signo de V_a es controlado directamente por τ , a través del circuito de conmutación en la Fig. (5.6). Así, la relación entre el torque de carga y el voltaje de armadura aplicado al motor aproximadamente es:

$$V_a(s) \approx \frac{K_r E}{T} \left[\frac{\Gamma(s) - \Gamma_L(s)}{s} \right]; \quad T \ll T_m \tag{5.22}$$

Vemos que el voltaje de armadura aplicado es proporcional a la integral del error de torque $\tau - \tau_L$, con constante de proporcionalidad $K_r E / T$. Ahora podemos combinar las ecuaciones (5.12), (5.13), (5.14) y (5.22) y resolver la función de

transferencia para el regulador de torque $G_r \triangleq \Gamma_L(s)/\Gamma(s)$, llegando al siguiente resultado:

$$G_r(s) \approx \frac{K_a K_r E (Js + B)}{NTs(Js + B)(L_a s + R_a) + NK_a K_b Ts + K_a K_r E (Js + B)} \quad (5.23)$$

El regulador de torque es un sistema de tercer orden con una ganancia DC unitaria en lazo cerrado cuando $s=0$. Frecuentemente la inductancia de armadura L_a suele ser muy pequeña y se puede despreciar. Haciendo $L_a = 0$ en la ecuación (5.23) y redefiniendo los coeficientes, se puede obtener una función de transferencia simplificada para el regulador de torque:

$$G_r(s) \approx \frac{K_1 (Js + B)}{s^2 + K_2 s + K_1 B}; \quad L_a \approx 0 \quad (5.24)$$

Aquí los parámetros $K_1 > 0$ y $K_2 > 0$ pueden ser calculados de los parámetros físicos originales del regulador de torque y del motor como sigue:

$$K_1 = \frac{K_a K_r E}{NR_a J T} \quad (5.25.a)$$

$$K_2 = \frac{NTR_a B + NTK_a K_b + K_a K_r E J}{NR_a J T} \quad (5.25.b)$$

De la ecuación (5.23), la ganancia DC del sistema de lazo cerrado es la unidad. Así, si $\tau(t)$ es constante, entonces en estado estacionario $\tau_L = \tau$, lo que significa que el torque entregado al eje q de la carga iguala al torque requerido τ . Esto se basa en la asunción que $0 \leq w < T$ y que la saturación ocurre cuando el ancho de pulso excede al periodo. La saturación de la señal *PWM* se produce cuando la magnitud del torque requerido excede al torque máximo τ_{max} , que puede ser entregado al eje de la carga. Un motor DC produce el torque máximo cuando el motor se “ahoga” y ya no gira; en

este caso θ_m y I_a son constantes. De las ecuaciones (5.7) y (5.8), vemos que la corriente de armadura es $I_{max} = E / R_a$, cuando se ahoga el motor donde E es el voltaje de alimentación DC. Así, de las ecuaciones (5.11) y (5.14), el torque máximo en el eje de la carga es:

$$\tau_{max} = \frac{K_a E}{R_a N} \quad (5.26)$$

El torque aplicado sobre el eje del motor debe satisfacer: $|\tau(t)| \leq \tau_{max}$ para evitar la saturación del circuito *PWM* del regulador de torque. En términos de la trayectoria de referencia $r(t)$, típicamente esto significa que la velocidad del robot a lo largo de la trayectoria deseada debe ser reducida, porque de lo contrario se requerirá mas torque de lo que el actuador es capaz de entregar.

Se puede demostrar que la función de transferencia del regulador de torque expresada por la ecuación (5.24) es estable sí y sólo sí $K_r > 0$. La ganancia K_r del regulador de torque es un parámetro designado que puede ser seleccionado por el usuario. La función de transferencia del regulador de torque se puede simplificar más si K_r es grande. De las ecuaciones (5.24) y (5.25), se obtiene:

$$\begin{aligned} \lim_{K_r \rightarrow \infty} G_r(s) &= \lim_{K_r \rightarrow \infty} \left[\frac{K_1(Js + B)}{s^2 + K_2s + K_1B} \right] \\ &= \lim_{K_r \rightarrow \infty} \left[\frac{K_1(Js + B)}{s^2 + JK_1s + K_1B} \right] \\ &= \lim_{K_r \rightarrow \infty} \left[\frac{Js + B}{s^2 / K_1 + Js + B} \right] \\ &= 1 \end{aligned} \quad (5.27)$$

Así para valores grandes de la ganancia K_r , el regulador de torque se puede modelar aproximadamente por sus características en estado estable $\tau_L = \tau$, asumiendo

que $|\tau| \leq \tau_{max}$. Esto es equivalente asumir que la dinámica del regulador de torque es rápida en comparación con el resto del sistema de la Fig. (5.3). En este caso podemos modelar el regulador de torque usando una ecuación algebraica, la cual lleve a su valor de saturación como:

$$\tau_L \approx \tau_{max} \text{sat}\left(\tau/\tau_{max}\right) \quad (5.28)$$

donde sat denota la función de *saturación unitaria* definida como:

$$\text{sat}(a) = \begin{cases} a & \text{si } |a| \leq 1 \\ 1 & \text{si } a > 1 \\ -1 & \text{si } a < -1 \end{cases}$$

El modelo de entrada/salida del regulador de torque se representa en la Fig. (5.8), donde el punto de ruptura τ_{max} se expresa en la ecuación (5.26).

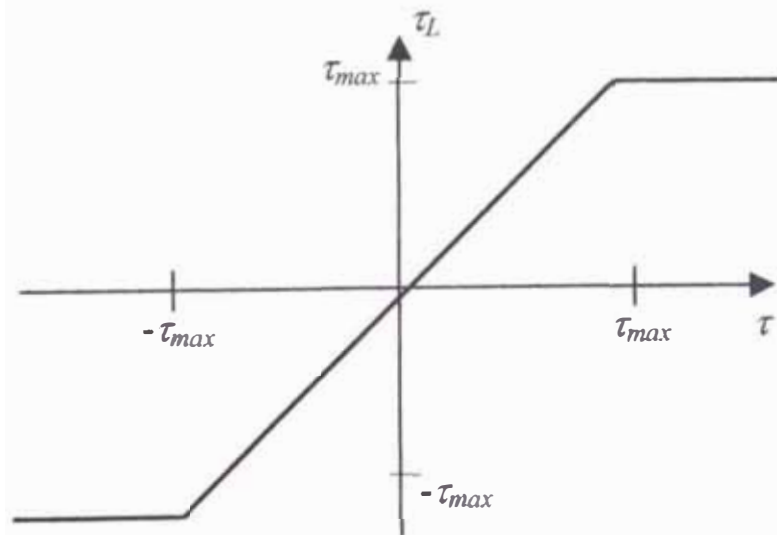


Fig. 5.8 Modelo de entrada/salida del regulador de torque.

Para concluir con el análisis, asumimos que K_r es suficientemente grande y que el regulador de torque se puede modelar teniendo en cuenta sus características en estado estacionario tal como se muestra en la Fig. (5.8). En la práctica, normalmente se asume que τ_{max} es suficientemente grande y que se puede ignorar.

- **Función de transferencia del controlador PID**

Se puede combinar el modelo simplificado del regulador de torque de la Fig. (5.8) con la función de transferencia del motor DC de la ecuación (5.15); asumiendo que τ_{max} es suficientemente grande y que se puede ignorar para la trayectoria de referencia de interés; el resultado es un modelo lineal simplificado de un controlador PID para una articulación simple tal, como se muestra en la Fig. (5.9).

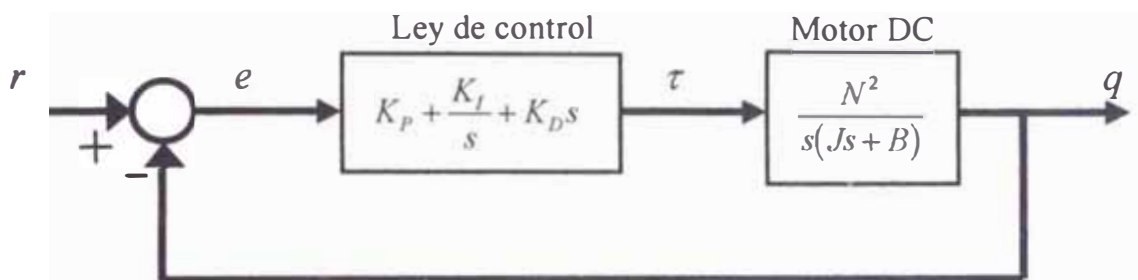


Fig. 5.9 Diagrama de bloques del controlador PID de articulación simple.

La función de transferencia del modelo de control mostrado en la Fig. (5.9) se hace teniendo en cuenta una serie de aspectos, asumiendo incluso que la carga incorporada en J y en B no cambia; en consecuencia, el modelo propuesto es estrictamente aplicable solamente a un robot de una sola articulación. Además, el eje de articulación debe considerarse alineado con el campo gravitacional, ya que no se ha incluido en el modelo del torque de la perturbación debido a la gravedad.

La función de transferencia $G_{PID}(s) \triangleq Q(s)/R(s)$ en lazo cerrado del controlador PID para una sola articulación mostrada en la Fig. (5.9) se puede expresada como:

$$G_{PID}(s) = \frac{N^2(K_D s^2 + K_P s + K_I)}{J s^3 + (N^2 K_D + B)s^2 + N^2 K_P s + N^2 K_I} \quad (5.29)$$

El controlador PID de una sola articulación se puede modelar como un sistema lineal de tercer orden. Evaluando $s=0$ en la expresión (5.29) de lazo cerrado se hace

igual a uno; en consecuencia, el error acarreado en estado estacionario para una entrada escalón unitario es cero, asumiendo que el sistema en lazo cerrado es estable.

La función de transferencia $G_{PID}(s)$ del sistema de lazo cerrado expresado por la ecuación (5.29) es estable sí y sólo sí los polos de $G_{PID}(s)$ se encuentran en el semiplano izquierdo del plano complejo. Usando el método indirecto denominado *prueba de Routh-Hurwitz* se puede determinar las condiciones que deben satisfacer las ganancias del controlador PID para que el sistema sea estable sin tener que hallar las raíces del polinomio característico. Los polos de $G_{PID}(s)$ de la ecuación (5.29) son los ceros del polinomio del denominador que no son canceladas por los ceros del polinomio del numerador. El polinomio $P(s)$ del denominador de $G_{PID}(s)$ es:

$$P(s) = Js^3 + (N^2K_D + B)s^2 + N^2K_Ps + N^2K_I \quad (5.30)$$

Con los coeficientes del polinomio $P(s)$ se construye la tabla de *Routh-Hurwitz* de la siguiente manera:

$$\begin{array}{lcl} s^3 & : & J \qquad \qquad \qquad N^2K_P \\ s^2 & : & N^2K_D + B \qquad \qquad N^2K_I \\ s^1 & : & \frac{(K_D + B)N^2K_P - JK_I}{K_D + B / N^2} \qquad 0 \\ s^0 & : & N^2K_I \end{array}$$

Para determinar las restricciones de las ganancias del controlador que garanticen la estabilidad del sistema se deben cumplir las condiciones necesaria y suficiente. Para satisfacer la condición necesaria, los coeficientes de $P(s)$ deben ser positivos, siendo $J > 0$; entonces se debe cumplir lo siguiente:

$$N^2K_D + B > 0, \quad N^2K_P > 0, \quad N^2K_I > 0 \quad (5.31.a)$$

Para satisfacer la condición de suficiencia los términos de la primera columna de

la tabla de *Routh-Hurwitz* deben ser diferentes de cero y del mismo signo; para esto se debe cumplir entonces lo siguiente:

$$N^2 K_D + B > 0, \quad (N^2 K_D + B)K_P - JK_I > 0, \quad N^2 K_I > 0 \quad (5.31.b)$$

De las condiciones (5.31) se concluye que las ganancias del controlador deben satisfacer lo siguiente:

$$K_P > 0 \quad (5.32.a)$$

$$K_I > 0 \quad (5.32.b)$$

$$K_D > -B/N^2 \quad (5.32.c)$$

$$(N^2 K_D + B)K_P > JK_I \quad (5.32.d)$$

Es de interés notar en la función de transferencia del controlador PID que si se hace $K_I = 0$ y/o $K_D = 0$, se puede aún mantener la estabilidad del sistema. Así se pueden obtener controladores P, PI y PD como casos especiales del controlador PID de la Fig. (5.9) que pueden ser usados potencialmente en el control de un robot. Las funciones de transferencia de estos controladores respectivamente serán:

$$G_P(s) = \frac{N^2 K_P}{J s^2 + B s + N^2 K_P} \quad (5.33.a)$$

$$G_{PI}(s) = \frac{N^2 (K_P s + K_I)}{J s^3 + B s^2 + N^2 K_P s + N^2 K_I} \quad (5.33.b)$$

$$G_{PD}(s) = \frac{N^2 (K_D s + K_P)}{J s^2 + (N^2 K_D + B) s + N^2 K_P} \quad (5.33.c)$$

Se puede comprobar que el error en estado estacionario del controlador PID para una entrada escalón y para una entrada tipo rampa es cero.

Una ventaja del controlador PID es que es muy simple de implementar; cada eje puede tener su propio lazo de control PID. El término de la ganancia proporcional al error inicializa la corrección en la señal de referencia del torque τ , sin embargo el error no es necesariamente cero. El término de la ganancia derivativa sirve para incrementar el margen de estabilidad. Finalmente el término de la ganancia integral tiende a reducir la magnitud del error acarreado en tiempo estacionario debido que cualquier error residual que puede aumentar es integrado, y en consecuencia, lleva en forma rápida a la corrección del torque τ .

La desventaja principal de controlador PID en robótica es que la carga vista por el motor o actuador de cada articulación puede variar rápida y sustancialmente. Esto es cierto particularmente en las articulaciones próximas a la base, donde los momentos de inercia y la carga debido a la gravedad pueden cambiar el orden de su magnitud. El análisis del controlador PID aquí presentado asume que la carga no cambia; en consecuencia, es estrictamente aplicable sólomente a un robot de una sola articulación, y que además el eje debe estar alineado con el campo de la gravedad que no se considera como una perturbación gravitacional. Así, el análisis del controlador PID es aplicable por ejemplo a los ejes verticales a la base de un robot multieje, si todas las otras articulaciones están fijadas como si el brazo condujera sólo enlaces rígidos. Si la carga sobre la articulación varía, se puede modelar como una perturbación, las cuales pueden ser compensadas por retroalimentación.

5.4.2 Control PD con compensación de la gravedad

Una de las deficiencias del controlador PID de articulación simple es que no tiene en cuenta los efectos de la gravedad. En esta sección se analizará un controlador tipo

PD que incluye un término explícito no lineal en la ley de control para compensar la gravedad y que nos puede servir para robots de n -ejes o articulaciones. Recalcando que las ecuaciones de estado de un manipulador robótico son de la siguiente forma:

$$\dot{q} = v \quad (5.34.a)$$

$$\dot{v} = \mathbf{D}^{-1}(q) [\tau - h(q) - c(q, v) - b(v)] \quad (5.34.b)$$

Se pueden controlar las variables de articulación indirectamente usando la formulación de retroalimentación de estado tal como se muestra en la Fig. (5.10). Aquí la variable τ que depende de q , v y r : $\tau = f(q, v, r)$ se llama *ley de control de retroalimentación de estado*. Nuestro objetivo es encontrar una ley de control f tal que la posición articular $q(t)$ del sistema de lazo cerrado siga estrechamente a la entrada de referencia $r(t)$.

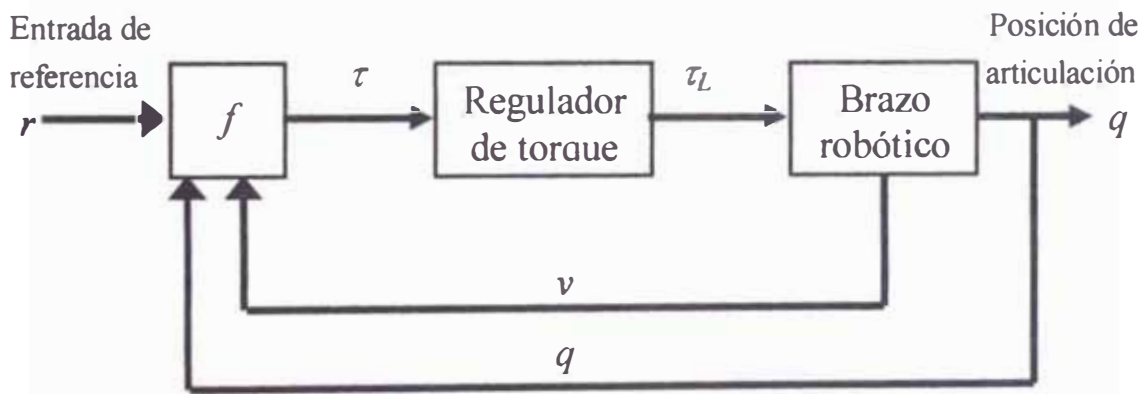


Fig. 5.10 Control de retroalimentación de estado de un brazo robótico.

El caso especial más simple pero tal vez el más importante ocurre cuando las entradas de referencia son $r(t) = a$ para algún punto constante de referencia $a \in \mathbf{R}^n$. A esto se le denomina *el problema de regulación*. Aquí debemos encontrar una ley de control que maneje al sistema desde una condición inicial arbitraria a la solución en estado estacionario $q(t)=a$; es decir, se desea obtener un controlador para regular

el brazo alrededor del valor de referencia a pesar de las perturbaciones que puedan ocurrir, tales como variaciones de la carga en el efector final del brazo.

La formulación proporcional-derivativa (PD) con compensación de la gravedad se puede considerar como una ley de control de retroalimentación de estado se puede escribir como:

$$e = r - q \quad (5.35.a)$$

$$\tau = K_P e + K_D \dot{e} + h(q) \quad (5.35.b)$$

Aquí K_P y K_D son matrices simétricas de $n \times n$ definidas positivas llamadas matriz de ganancia de posición y matriz de ganancia de velocidad, respectivamente. De la ecuación (5.35.b) vemos que el torque τ consiste de términos proporcionales al error y a la derivada del error, más un término no lineal de la gravedad $h(q)$. Si se sustituye la ecuación (5.35) en la ecuación (5.34), se obtiene la ecuación de movimiento de lazo cerrado para el sistema de la Fig. (5.10) como:

$$\dot{q} = v \quad (5.36.a)$$

$$\dot{v} = \mathbf{D}^{-1}(q) [K_P(r - q) + K_D(\dot{r} - v) - c(q, v) - b(v)] \quad (5.36.b)$$

Suponiendo que $r(t) = a$ para $t \geq 0$ para algún punto de referencia a , entonces para analizar los puntos de equilibrio resultantes del sistema de lazo cerrado, se igualan a cero los términos del lado derecho de la ecuación (5.36) y se resuelve para q y v resultando:

$$v = 0$$

$$\mathbf{D}^{-1}(q) [K_P(a - q) - c(q, 0) - b(0)] = 0$$

Teniendo en cuenta las ecuaciones (4.34) y (4.24) entonces $c(q, 0) = 0$ y $b(0) = 0$; así, $K_P(a - q) = 0$, donde K_P es una matriz definida positiva. En consecuencia, el

sistema de lazo cerrado expresado en la ecuación (5.36) tiene un punto simple de equilibrio en:

$$\hat{x}^T = [a^T, 0^T] \quad (5.37)$$

Se debe notar que $x(t) = \hat{x}$ es la solución deseada en tiempo estacionario del brazo cuando la entrada de referencia $r(t) = a$; por consiguiente, para demostrar que el controlador PD con compensación de la gravedad tiene un comportamiento deseado en lazo cerrado, es suficiente demostrar que el punto de equilibrio \hat{x} es asintóticamente estable y que su dominio de atracción rodea al espacio de estado en forma completa.

Para demostrar que $x^T = [q^T, v^T]$ es la solución del brazo robótico de la ecuación (5.34), se asume que el torque τ es calculado mediante la ley de control expresada por la ecuación (5.35) con $r(t) = a$ para $t > 0$. El punto de equilibrio $\hat{x}^T = [a^T, 0^T]$ es asintóticamente estable y el dominio de atracción será $\Omega = \mathbf{R}^{2n}$; así que para cada $x(0) \in \mathbf{R}^{2n}$: $x(t) \rightarrow \hat{x}$ cuando $t \rightarrow \infty$. Para hacer la demostración se puede usar el segundo método de *Liapunov* visto en el anexo D.

Si los efectos de la gravedad han sido eliminados, entonces la ecuación del brazo robótico no lineal se puede controlar suficientemente usando un simple controlador lineal PD. La única restricción para que el sistema de lazo cerrado sea estable, es que las ganancias del controlador proporcional K_P y derivativa K_D deben ser matrices definidas positivas y en forma especial se pueden elegir matrices diagonales. En la Fig. (5.11) se muestra el esquema de diagrama de bloques del controlador PD con compensación de la gravedad para un robot manipulador.

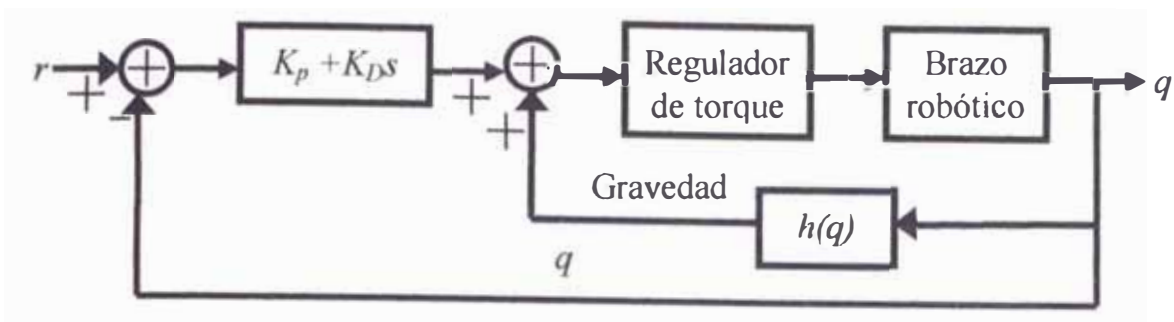


Fig. 5.11 Controlador PD con compensación de la gravedad.

El significado práctico de la ley de control PD con compensación de la gravedad en la ecuación (5.35) consiste en el hecho que no requiere un conocimiento detallado del tensor inercia $D(q)$ del manipulador del vector de acoplamiento de velocidades centrífugo y de Coriolis $c(q, v)$, o del vector de fricción $b(v)$ sólo se requiere del conocimiento del vector de carga por efecto de la gravedad $h(q)$, y esto es relativamente fácil determinar, al menos en comparación con los otros efectos. En una implementación real de la ecuación (5.35), el valor exacto de $h(q)$ no puede ser evaluado en forma precisa por lo que $h(q)$ se puede escribir como:

$$h(q) = h_e(q) + \Delta h \quad (5.38)$$

donde el término $h_e(q)$ es un valor estimado de $h(q)$ y Δh representa el error en la estimación de $h(q)$ causado por la incertidumbre de la masa exacta y la geometría de los enlaces del robot. Usando $h_e(q)$ en lugar de $h(q)$ en la ley de control se realizará la ejecución del controlador en forma aproximada. Si por ejemplo, cuando se presente una pequeña variación en la localización del punto de equilibrio, lo cual significa que error acarreado en tiempo estacionario en la respuesta ante una entrada escalón no llegará a hacerse exactamente igual a cero.

Es de interés comparar el controlador no lineal PD con compensación de la gravedad para un robot de n -ejes representado en la Fig. (5.11), en el sentido que este

controlador puede ser considerado como una consecuencia teórica del método de control lineal PID. Para ver esto, suponemos que la elección de las matrices de las ganancias en la ecuación (5.35) son definidas positivas; para esto se pueden elegir matrices diagonales con sus elementos positivos:

$$K_P = \text{diag}\{k_{P1}, k_{P2}, \dots, k_{Pn}\} \quad (5.39.a)$$

$$K_D = \text{diag}\{k_{D1}, k_{D2}, \dots, k_{Dn}\} \quad (5.39.b)$$

Si consideramos un eje k para un robot en el cual $h_k(q) = 0$, entonces la ley de control PD con compensación de la gravedad es idéntica a la ley de control PID de simple articulación cuando $K_I = 0$, tal como se muestra en la ecuación (5.33.c), entonces podemos ver que las restricciones en las ganancias del controlador se reducen a $K_P > 0$ y $K_D > -B/J$, y las restricciones correspondientes al controlador del eje k serán $K_{Pk} > 0$ y $K_{Dk} > 0$. De la condición suficiente vista en la ecuación (5.31) que se acondiciona sobre las fronteras conservativas en las ganancias del controlador se puede asumir que la fricción $b(v)$ es despreciable. Entonces las restricciones vienen a ser idénticas.

5.4.3 Control mediante el método del torque calculado

El método basado en la ley de control de *Torque calculado*, se basa en las ecuaciones dinámicas de movimiento del robot expresadas por el algoritmo *Lagrange-Euler*, que consiste en hallar torques o fuerzas apropiadas para controlar todas las articulaciones del robot en tiempo real de tal manera que se siga una trayectoria temporal muy estrechamente cercana como sea posible a la deseada. El torque de cada motor requerido para controlar al robot se basa en el modelo dinámico

del mismo. Las características de tensión o corriente del motor se modelan también en el esquema de cálculo de la dinámica y el *torque calculado* se convierte en la corriente aplicada al motor a través de un regulador de torque. La tensión aplicada se da en forma de modulación de ancho de pulso a una frecuencia adecuada para que los efectos de muestreo se ignoren en el análisis. Debido a los errores de modelamiento y a las variaciones de los parámetros en el modelo dinámico, se utilizan las señales de retroalimentación de posición y velocidad para calcular los torques de corrección, los cuales, cuando se suman a los torques calculados basados en el modelo del robot, proporcionan la señal de excitación correctora para los motores de las articulaciones.

La técnica del *torque calculado* básicamente es un control anticipativo o estimado que contiene componentes directas y en retroalimentación. Las componentes de control compensan las fuerzas de interacción entre todas las articulaciones y las componentes de retroalimentación permiten la corrección necesaria para compensar cualquier desviación de la trayectoria deseada.

Recalcando que el *problema de control de regulación* corresponde al caso cuando la entrada de referencia $r(t)$ es constante, pero el caso más general se da cuando $r(t)$ es variable. Este es el llamado el *problema de control de seguimiento* o *servocontrol*. Asumiendo que la trayectoria planificada $r(t)$ es suficientemente suave y que tiene al menos dos derivadas, la técnica propuesta para resolver el problema de seguimiento del robot es el llamado *Método de Torque calculado*, el cual hace uso directo del modelo dinámico completo del manipulador expresado por la ecuación (5.1) para cancelar no solamente los efectos de la gravedad, sino también las fuerzas de los efectos de Coriolis y centrífuga, el efecto inercial del manipulador y la fuerza de fricción, minimizando los efectos no lineales de modo tal que se pueda utilizar un

regulador proporcional- derivativo PD, y así controlar los motores de articulación del robot.

Para la implementación del método de control de *Torque calculado* se usa una ley de control con una dinámica lineal deseada que nos gustaría que el robot tuviera se puede expresar de la siguiente manera:

$$\ddot{q} = \ddot{q}_r + K_p(q_r - q) + K_v(\dot{q}_r - \dot{q}) \quad (5.40)$$

donde:

- q, \dot{q}, \ddot{q} son respectivamente los vectores del desplazamiento, de la velocidad y de la aceleración reales de las articulaciones del robot.
- $q_r, \dot{q}_r, \ddot{q}_r$ son respectivamente los vectores del desplazamiento, de la velocidad y de la aceleración de referencia de las articulaciones del robot.
- K_p y K_v son matrices simétricas de $n \times n$ definidas positivas asociados con las ganancias de los términos de retroalimentación de posición y de velocidad de las articulaciones del robot respectivamente.

Si se expresan los términos del modelo dinámico original en lazo abierto del robot visto en la expresión (5.1) mediante los valores estimados de estos términos con los respectivos errores producidos durante la estimación obtenemos:

$$D(q) = D_e(q) + \Delta D \quad (5.41.a)$$

$$c(q, \dot{q}) = c_e(q, \dot{q}) + \Delta c \quad (5.41.b)$$

$$h(q) = h_e(q) + \Delta h \quad (5.41.c)$$

$$b(\dot{q}) = b_e(\dot{q}) + \Delta b \quad (5.41.d)$$

donde:

- $D_e(q), h_e(q), c_e(q, \dot{q}), b_e(\dot{q})$ denotan los valores estimados de los términos del

efecto inercial, fuerzas debido al efecto de Coriolis y centrífuga, fuerzas gravitatorias y fuerzas de fricción respectivamente.

- ΔD , Δc , Δh , Δb denotan los errores respectivos al estimar los términos del modelo dinámico.

La técnica del *torque calculado* se puede apreciar en la Fig. (5.12) en un diagrama de bloques en el cual la mitad derecha del diagrama intenta cancelar la dinámica no lineal indeseada, mientras que la mitad izquierda inserta la dinámica lineal deseada, suponiendo los parámetros estimados del robot son tratados como si fueran reales.

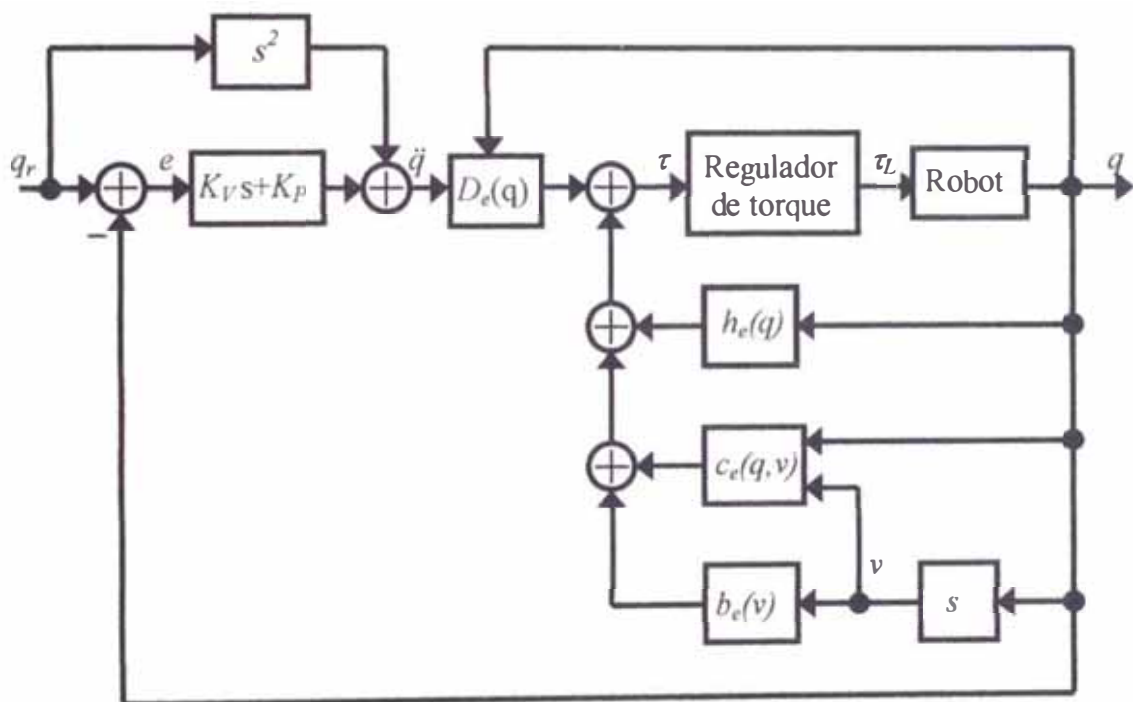


Fig. 5.12 Diagrama de bloques del control de torque calculado.

Para cancelar los términos no lineales de la expresión (5.1) del modelo dinámico del manipulador, se puede usar una ley de control de retroalimentación de estado que se obtiene reemplazando la ecuación (5.40) y los términos estimados de expresión (5.41) en la ecuación (5.1) de la siguiente manera:

$$\tau = h_e(q) + c_e(q, v) + b_e(v) + D_e(q) \left[K_p e + K_v \dot{e} + \ddot{q}_r \right] \quad (5.42)$$

siendo $e = q_r - q$ es el vector de error de desplazamiento de articulación del sistema de control en lazo cerrado.

Hay que notar que la estimación de todos los términos del modelo dinámico en lazo abierto aparecen en la ley de control en retroalimentación expresado en la ecuación (5.42) y esquematizado en la Fig. (5.12).

La solución del robot en la ecuación (5.1) es $x^T = [q^T, v^T]$. Asumiendo que el torque τ es calculado usando la ley de control en la ecuación (5.42), entonces se puede examinar los efectos de la ley de control formulando una ecuación diferencial de segundo orden de la dinámica del robot en lazo cerrado en términos del vector de error e . Combinando las ecuaciones de estado del torque τ de lazo abierto del robot manipulador de la ecuación (5.2) con la expresión (5.42); luego de realizar operaciones y simplificar se llega a obtener:

$$D(q) \left(\ddot{e} + K_v \dot{e} + K_p e \right) = \Delta D \left(K_p e + K_v \dot{e} + \ddot{q}_r \right) + \Delta h + \Delta c + \Delta b \quad (5.43)$$

Es de interés analizar qué pasa cuando los términos estimados de la dinámica del robot son exactos; es decir, si todos si los Δ 's de la ecuación de error (5.43) son muy pequeños. En este caso ideal los términos no lineales son completamente cancelados y vemos que la ecuación de error se reduce a una ecuación lineal de segundo orden *independiente* de los parámetros del robot, del siguiente modo:

$$D(q) \left(\ddot{e}(t) + K_v \dot{e}(t) + K_p e(t) \right) = 0 \quad (5.44)$$

Como la matriz $D(q)$ es siempre no singular, entonces analizamos sólo los términos lineales que están dentro del paréntesis; por tanto los términos no lineales

son completamente cancelados. Si las matrices de ganancia K_P y K_V son diagonales, entonces las ecuaciones de movimiento en lazo cerrado no son solamente lineales, sino que también se encuentran desacopladas una de la otra. Aplicando la transformada de *Laplace* a la ecuación (5.44) obtenemos la expresión del error como:

$$E(s) = \frac{s e(0) + \dot{e}(0) + K_v e(0)}{s^2 + K_v s + K_p} \quad (5.45)$$

Si a las matrices K_P y K_V las elegimos convenientemente para que las raíces del polinomio característico en la ecuación (5.45) tengan partes reales negativas de manera que el vector de error de posición $e(t)$ tienda asintóticamente a cero, y además hacemos que las matrices K_P y K_V sean diagonales; entonces se tiene un sistema de control desacoplado, pudiéndose controlar cada motor de articulación independientemente de los otros. Por tanto, la ecuación característica del error de desplazamiento de la expresión (5.45) para el motor k será :

$$s^2 + K_v^k s + K_p^k = 0 \quad (5.46)$$

Haciendo la analogía de la expresión (5.46) con la ecuación característica general de un sistema de segundo orden dada por la ecuación:

$$s^2 + 2 \zeta w_n s + w_n^2 = 0 \quad (5.47)$$

Luego, igualando e identificando términos de las ecuaciones (5.46) y (5.47) se puede obtener los valores de las ganancias del controlador PD para cada articulación del robot como sigue:

$$K_p^k = (w_n^k)^2 \quad \text{y} \quad K_v^k = 2 \zeta w_n \quad (5.48)$$

El comportamiento de un sistema de segundo orden está dictado por su frecuencia natural no amortiguada w_n y por el coeficiente de amortiguamiento ζ . Por razones de

seguridad, el sistema no puede tener una respuesta subamortiguada para una entrada de posición articular tipo escalón, sino que debe ser un sistema con respuesta críticamente amortiguada o sobreamortiguada, por lo que se requiere entonces que $\zeta > 1$. Con el fin de no excitar la oscilación estructural y la resonancia de la articulación, la frecuencia natural no amortiguada w_n se puede fijar en: $w_n \leq 0.5 w_r$, siendo w_r la frecuencia resonante estructural de la articulación la cual es una propiedad del material utilizado en la construcción del manipulador.

En conclusión, el tratamiento de linealidad y desacoplabilidad es aceptable cuando el robot se mueve a bajas velocidades; pero cuando el manipulador se está moviendo a velocidades altas, el control usando la técnica del *torque calculado* en tiempo real es sumamente difícil o casi imposible, a menos que se cuente con un computador de una gran potencia numérica que procese los términos estimados de la dinámica a velocidades significativamente grandes.

Existe un número de desventajas prácticas del método de control de *torque calculado*. La primera por su puesto es que los términos estimados del robot nunca son exactos; la estimación con frecuencia contiene errores significativos porque muchos de los parámetros individuales no se pueden medir directamente, sino se deduces de otras mediciones. Otra deficiencia que impide la cancelación completa de los efectos no lineales, es que el modelo dinámico por sí mismo nunca es completo. Por ejemplo, no se tiene en cuenta la flexibilidad en los enlaces, el juego en los engranajes ó la carga en el extremo final de brazo generada por el objeto que se está manipulando. Finalmente si la entrada de referencia $r(t)$ varía muy rápidamente, entonces generará términos en el torque $\tau(t)$ cuyos valores exceden los límites de los actuadores de articulación que pueden causar saturación en el regulador de torque.

5.5 Control del *ROB-DEX*

El *ROB-DEX* tiene una estructura mecánica totalmente rígida en forma vertical y flexible en forma horizontal por lo que se puede hacer uso de cualquier de los métodos de control descritos en este capítulo. De la cinemática directa e inversa del *ROB-DEX* dada por las ecuaciones (2.25) y (2.33) respectivamente y de la expresión de la dinámica del *ROB-DEX* denotada por la expresión (4.53), podemos ver que las tres primeras articulaciones son angulares y están acopladas entre sí, mientras que la cuarta articulación es prismática y está totalmente desacoplada de las demás articulaciones. Esta característica de configuración del *ROB-DEX* hace más sencillo el análisis de su cinemática, su dinámica y se puede aplicar eficientemente cualquier método o estrategia de control.

El funcionamiento y control del *ROB-DEX*, consiste en el procesamiento numérico en tiempo real de los modelos matemáticos de cinemática, planificación de trayectoria, la dinámica y el algoritmo de control, con la finalidad de obtener el desplazamiento y la velocidad deseados, así como la trayectoria programada, usando al computador como un gran controlador digital.

Si las articulaciones del *ROB-DEX* estuvieran accionadas con motores DC, se pueden usar el método de control PID de simple articulación para las tres primeras articulaciones que son rotacionales, debido a que su modelo dinámico de la ecuación (4.53) no se ve afectado por el efecto gravitatorio solamente por el efecto inercial; las fuerzas debido al acoplo de velocidades y al efecto de las fuerzas de fricción. Por tanto, si el robot se mueve a velocidades moderadas, las fuerzas debido al efecto de acoplo de velocidades prácticamente desaparecerían, permaneciendo sólo las fuerzas debido únicamente al efecto inercial de los enlaces del robot. Para la cuarta

articulación prismática se puede utilizar el método de control PD con compensación de la gravedad como un término constante y sin acoplo de velocidades de ningún tipo. En el modelo dinámico del *ROB-DEX* se puede notar que la cuarta articulación es la única que está afectada por la gravedad.

El control del *ROB-DEX* usando la combinación de las técnicas de control indicadas anteriormente se puede apreciar en un diagrama de bloques completo en la Fig. (5.13) en el cual las variables con (*) significan señales digitalizadas, considerándose en este esquema además los bloques A/D (Conversores Analógico a Digital) y D/A (Conversores Digital a Analógico). En la Fig. (5.13) fácilmente se distingue que las variables y parámetros del *ROB-DEX* son:

- θ_r = vector articulación angular de referencia que representa a las tres primeras articulaciones.
- w_r = vector velocidad de referencia de las tres primeras articulaciones.
- d_{4r} = desplazamiento de la cuarta articulación prismática.
- v_{4r} = velocidad de la cuarta articulación.
- q = vector 4×1 que representa la posición de las articulaciones reales.
- v = vector 4×1 que representa las velocidades reales.
- e_q = vector del error de articulación angular de las tres primeras articulaciones.
- e_w = vector del error de velocidad angular de las tres primeras articulaciones.
- e_d = error de articulación lineal de la cuarta articulación.
- e_v = error de velocidad lineal de la cuarta articulación.
- θ_m = vector 4×1 que representa los desplazamientos angulares de los motores que accionan las articulaciones.

- K_p = matriz diagonal que representa las ganancias proporcionales al error de las tres primeras articulaciones.
- K_I = matriz diagonal que representa las ganancias integrales del error de las tres primeras articulaciones.
- K_D = matriz diagonal que representa las ganancias derivadas del error de las tres primeras articulaciones.
- k_p = ganancia proporcional al error de la cuarta articulación.
- k_D = ganancia derivativa del error de la cuarta articulación.
- y = vector que representa la configuración de la herramienta.

En caso que se requiera considerar un método de control más completo y realista para el *ROB-DEX*, en el cual se considere todos los términos no lineales del modelo dinámico visto en la ecuación (4.52), entonces se podría implementar el método de control del *torque calculado*. Un esquema detallado de este método se muestra en la Fig. (5.14), en el cual las entradas de referencia digitalizadas son la posición (q_r^*), la velocidad (v_r^*) y a la aceleración (a_r^*) de cada articulación del *ROB-DEX*.

La ley de control de realimentación de estado en la ecuación (5.40) se define como:

$$a = a_r + K_p(q_r - q) + K_v(v_r - \dot{q}) \quad (5.49)$$

donde de la ecuación (5.49) y de la Fig. (5.14) se tienen:

- q, v, a son los vectores 4×1 del desplazamiento, de la velocidad y de la aceleración reales de las articulaciones respectivamente.
- q_r, v_r, a_r son los vectores 4×1 del desplazamiento, de la velocidad y de la aceleración de referencia de las articulaciones respectivamente.

- q^* , v^* , a^* son los vectores 4×1 del desplazamiento, de la velocidad y de la aceleración reales digitalizados respectivamente de las articulaciones.
- q_r^* , v_r^* , a_r^* son los vectores 4×1 digitalizados del desplazamiento, de la velocidad y de la aceleración de referencia de las articulaciones respectivamente.
- K_P y K_V son matrices simétricas de 4×4 definidas positivas y diagonales que denotan las ganancias de los términos de retroalimentación de estado de posición y de velocidad de las articulaciones respectivamente.
- $y(q)$ es el vector de configuración de la herramienta que indica su localización en función del vector articular dado por la cinemática en la ecuación (2.25).

Como los actuadores que impulsan al *ROB-DEX* son motores paso a paso, y éstos pueden trabajar en lazo abierto, tan sólo bastaría los algoritmos de la cinemática y la planificación de la trayectoria, y no será necesario desarrollar ninguna estrategia ni tipo de control con retroalimentación, sino usar un método simple de control en lazo abierto. Pero durante la operación del *ROB-DEX*, se presenta una acumulación de errores en forma de perturbaciones que pueden ser debido al diseño de la estructura mecánica, a vibraciones de sus elementos, a su sistema de transmisión, etc., lo que produce desviaciones en su posición y velocidad durante el movimiento; entonces, para subsanar este inconveniente y alcanzar los objetivos pre-establecidos, se usa un algoritmo de control de lazo cerrado con la retroalimentación de posición y velocidad angulares para controlar los motores de paso, con la información tanto de los valores de referencia y los valores reales registrados por los sensores respectivos.

El método usado para controlar el *ROB-DEX* en forma completa se muestra en la Fig. (5.15) en forma de diagrama de bloques. El control se realiza físicamente sin tener en cuenta el modelo dinámico en forma instantánea porque los motores se han

sobredimensionado y considerando que el *ROB-DEX* trabaja a muy bajas velocidades.

En el diagrama de bloques de la Fig. (5.15) se puede apreciar que las magnitudes de referencia digitalizadas son el vector de articulación (q_r^*) y el vector de velocidad articular (v_r^*); ambas magnitudes son divididas entre sus respectivas relaciones de engranajes (N) para obtener los vectores de la posición angular (q_{mr}^*) y de velocidad (v_{mr}^*) respectivamente como señales de control de los motores que accionan las articulaciones del robot. Estos valores son comparados con los valores reales digitalizados de la posición (q_m^*) y velocidad (v_m^*) angulares de los motores en forma respectiva, obteniéndose las señales de error tanto de la posición (e_q^*) y de la velocidad (e_v^*). Las señales de error se utilizan para generar un tren de impulsos, los que serán usados como señales tipo reloj en los controladores de los motores de paso. El signo del error es usado para determinar el sentido de giro de cada motor. La señal de error de velocidad se utiliza para determinar la frecuencia del tren de impulsos del error de posición.

El controlador de motor de paso genera cuatro señales (Q_1, Q_2, Q_3, Q_4) para cada motor. Estas señales se amplifican mediante transistores de potencia con el objetivo de proporcionar la energía que necesita cada motor, obteniéndose de esta manera las señales ($Q_{1L}, Q_{2L}, Q_{3L}, Q_{4L}$), las cuales son utilizadas para energizar las bobinas de los motores de paso.

Cada motor de paso del *ROB-DEX* desarrolla en su respectivo eje de giro, un desplazamiento angular (θ_m) y una velocidad (V_m), las cuales serán sensadas mediante codificadores ópticos relativos para la posición y mediante tacómetros digitales para la velocidad que sirven para cerrar el lazo de control. Para obtener el

desplazamiento real de los eslabones del *ROB-DEX* se utilizan engranajes reductores para las tres primeras articulaciones y un tornillo sin fin para la cuarta articulación. Las magnitudes de estas relaciones son $(1/20, 1/10, 5/9, 9/40\pi)$ respectivamente en cada articulación.

Cuando se han alcanzado los valores reales de articulación q se puede obtener entonces la localización $y(q)$ real del robot utilizando la ecuación (2.25) de la cinemática directa del *ROB-DEX*.

La estrategia de control explicada en la Fig. (5.15) nos permite evaluar, monitorear y conocer el torque en base a las magnitudes realimentadas de la posición y velocidad instantáneas y compararlo luego con el torque del motor de paso, consistenciando el programa de control para que no supere al torque máximo desarrollado por cada motor, y así el robot pueda trabajar normalmente.

CAPITULO VI

PROGRAMACION DEL ROBOT

6.1 Introducción

El hecho de definir a un robot como una máquina programable, implica que el sistema de control debe estar capacitado para adquirir la información necesaria acerca de cómo ejecutar una tarea, por lo que la programabilidad es una característica explícita para suministrar al robot dicha información para su operación.

La programación dota a los robots de numerosas ventajas sobre los manipuladores cableados convencionales, entre las que cabe destacar la posibilidad de adecuarlos con relativa facilidad a las especificaciones de una aplicación concreta.

Para ser programables se suele incorporar uno o varios procesadores de propósito general; los lenguajes; en cambio, son específicos y distintos en numerosos aspectos de los habituales en programación de computadores, debido básicamente a las siguientes circunstancias:

- El entorno en que actúa el robot generalmente no puede describirse en términos puramente cuantitativos.
- Es necesario incluir condiciones no usuales, tales como la prevención de colisiones.
- Las acciones del robot están sujetas a imprecisiones que pueden dar lugar a incidentes que el programa debe ser capaz de tratar.

- Ciertas informaciones sensoriales del sistema pueden ser no sólo difíciles de procesar en tiempo real sino también ambiguas.

Así por ejemplo, la programación genérica de una operación de ensamblaje requiere resolver de forma general aspectos tales como:

- La descripción de las distintas piezas.
- Especificar el plan de ensamblaje, que depende de la configuración del robot.
- Determinar la orientación de la pieza, así como de los puntos y posición de prensado.
- Determinar trayectorias libres de colisiones en todos los desplazamientos.
- Suministrar los parámetros necesarios a las diferentes rutinas de ensamblaje.
- Definir las estrategias a seguir en el caso probable de que se produzcan errores y determinar las acciones a tomar en caso de fallos.

En numerosas aplicaciones industriales, los robots se programan guiando el brazo según la secuencia de operaciones necesarias para la aplicación que posteriormente será repetida idénticamente en forma cíclica, siendo imposible la utilización de sensores y la modificación del programa debido, respectivamente, a la ausencia de definición de acciones condicionales y de texto.

La programación a nivel robot introduce variables cuyo valor depende de la información adquirida en tiempo de ejecución, y que puede regir la toma de decisiones; introduce estructuras de control que habilitan acciones condicionadas a la información de los sensores y facilita la interactividad en tiempo de programación.

La programación a nivel tarea, dá un paso más hacia la abstracción, al permitir especificar las operaciones desde el punto de vista de relaciones de posición entre las partes, prescindiendo de las configuraciones del robot.

6.2 Programación no textual

La programación no textual de un robot se da cuando éste interviene en el proceso de programación actuando como un digitalizador de posiciones de n dimensiones, posiciones por la que luego pasará en la fase de ejecución. También se le denomina programación por guiado.

La programación de robots a distintos niveles jerárquicamente superiores se apoya en lo que podríamos llamar *programación de mando o elemental*, cuyo similar en programación de computadores sería el ensamblador.

De forma genérica, la programación por guiado presenta la ventaja de que un operador sin experiencia y sin necesidad de emplear símbolos abstractos, pueda hacer un programa, pero adolece de la limitación de que éste queda forzosamente constituido por una secuencia fija de movimientos, sin posibilidad de modificaciones ni de relación inteligente con el medio a través de sensores. Por otra parte, en sí misma, es ineficiente desde el punto de vista del programador, cuyo trabajo es tedioso e incluso peligroso, y por la infrautilización del robot que no realiza trabajo útil mientras es programado.

Atendiendo a la forma en que se fijan los puntos de paso de la trayectoria del robot, pueden distinguirse dos grandes tipos de guiado:

- *Guiado pasivo.*- Cuando el operador posiciona el robot con sus propias manos. En este modo sólo se tiene acceso a las coordenadas articulares que luego tienen que ser convertidas a tensiones de mando de los servomotores.
- *Guiado activo.*- En que el robot es posicionado desde un puesto de mando (teclado, “joystick” u otros) teniéndose acceso en curso de enseñanza a las tensiones de mando que pueden memorizarse de forma continua mediante

registradores magnéticos multicanal o de forma discreta muestreándolas mediante un computador digital que las memoriza.

En los sistemas de programación por guiado puede hacerse una taxonomía atendiendo a dos aspectos: cómo se especifican las posiciones del robot y cual es el repertorio de movimientos entre las posiciones.

6.3 Programación a nivel de robot

Un sistema de programación es de nivel robot (o explícita) cuando la secuencia de configuraciones del robot es definida por un programa que puede contener sentencias de especificación de movimiento y relativas a información del entorno.

Este nivel, en el que se agrupa la mayor parte de los lenguajes desarrollados, es aun muy próximo al robot, siendo común en él que el programador pueda posponer a la ultima fase de elaboración del programa la entrada de ciertos parámetros, tales como pares, trayectorias u otros, a fin de introducirlos usando los sensores del propio brazo como transductores de entrada. En otros términos, la secuencia se establece en el programa y ciertas posiciones se definen por guiado.

En aquellos lenguajes de este grupo en que existe una librería de funciones de nivel inferior, es también el programador quien tiene que seleccionar las funciones necesarias y su secuencia, de acuerdo con la tarea a ejecutar y las restricciones del entorno.

Los lenguajes a nivel robot están orientados a la descripción de las operaciones que debe realizar el robot y de esta concepción nacen sus dos aspectos más positivos: posibilidad de especificar distintos tipos de movimientos y trayectorias y capacidad de interacción con el entorno.

6.3.1 Ventajas e inconvenientes

En estos lenguajes de programación es usual disponer de varias primitivas de movimientos: de poca precisión, de precisión, rectilíneo o circular entre puntos en el espacio cartesiano, rectilíneo en el espacio articular.

Poder utilizar información sensorial impone al menos dos requisitos a la programación: disponer de mecanismos generales de entrada-salida así como de estructuras de control versátiles.

El programador podrá hacer uso de esta información para:

- Iniciar y terminar movimientos: sincronización con otros elementos de producción, movimientos precavidos.
- Escoger entre acciones alternativas.
- Identificar objetos y sus posiciones.
- Acomodarse a condiciones externas de forma activa: movimientos acomodativos.

Finalmente, algunos sistemas permiten descomponer el programa en un módulo de control del proceso y otro de control del robot.

Los principales inconvenientes provienen de los requisitos que se imponen al programador:

- Tiene que ser un conocedor de las técnicas de programación de computadores.
- Tiene que tener experiencia en la concepción de estrategias de movimiento basadas en información sensorial.
- Tiene que prever cualquier incidente posible, y su tratamiento asociado.
- Los programas, incluso para tareas simples, requieren un buen número de instrucciones.

6.3.2 Consideraciones respecto al diseño informático

En los lenguajes a nivel robot se aprecian cuatro tendencias en cierto modo contrapuestas dos a dos: compiladores y traductores por una parte y lenguajes nuevos respecto a ampliaciones de lenguajes de programación de computadores.

Compiladores e intérpretes.- Los lenguajes compiladores permiten alcanzar una gran eficiencia durante la fase de explotación de los programas manteniendo una suficiente capacidad de detección de errores de programación. Sin embargo, la corrección de errores detectados en línea, en fase de depuración, necesita de mecanismos en muchos casos excesivamente largos.

Los intérpretes permiten entornos de programación más interactivos ante todo en depuración y se pueden ampliar más fácilmente. Estos aspectos parecen haber pesado más que los aspectos positivos de los lenguajes compilados, ya que la mayoría de sistemas a nivel robot están basados en intérpretes.

Ampliaciones.- De hecho, para diseñar un lenguaje no existen más que dos vías: ampliar uno de propósito general ya existente o crear uno nuevo. La primera solución, adoptar uno de propósito general, resulta atractiva por minimizar el tiempo de aprendizaje necesario por parte de los futuros programadores cuando éstos ya conocen el lenguaje de programación utilizados como base.

Crear nuevos lenguajes tiene la ventaja de minimizar el requisito de que el programador sea un conocedor de las técnicas de programación de computadores. Si además en el diseño se pone especial énfasis en especializar la interfaz usuario-sistema para tareas de robótica, es posible que el sistema sea programado por técnicas provenientes de control numérico.

La única razón de peso para acometer el diseño de un nuevo lenguaje radica en la

necesidad que tienen los sistemas de programación fuera de línea de robots, de entornos específicos de los que, generalmente, no se dispone en lenguajes de programación de computadores.

6.4 Programación a nivel tarea

Aunque actualmente la programación industrial de robots se hace a nivel manipulador, diversos centros de investigación están trabajando desde hace unos años en lenguajes de muy alto nivel, en que el usuario pueda especificar las tareas en términos de relaciones de posición entre las piezas a ensamblar, en lugar de expresarlas en función de configuraciones sucesivas del manipulador.

Estos sistemas de programación, todos ellos aún en fase de experimentación, requieren de modelos geométricos completos del entorno de trabajo y son conocidos como *sistemas con modelo del entorno*.

La comparación de estos sistemas con los del nivel robot conduce a las siguientes conclusiones:

- La especificación de una operación a nivel tarea es completamente independiente del robot utilizado. El usuario no tiene que declarar explícitamente ninguna posición o trayectoria que dependa de la dinámica o cinemática del robot.
- En un sistema nivel tarea, la generación de las primitivas de los movimientos elementales que es necesario que realice el robot para llevar a cabo la operación es automática y se hace a partir de las relaciones espaciales entre los objetos especificados en la descripción de la tarea.
- A partir de una cierta complejidad de las tareas a realizar, éstas se describen más fácilmente con este tipo de lenguajes que con los explícitos.

- Si bien no presentan los inconvenientes de los lenguajes explícitos, en el caso compartido necesitan de computadores más potentes.
- Los lenguajes que incorporan modelos del entorno son capaces de efectuar tareas más complejas y cuentan en su repertorio con instrucciones más abstractas.
- Cuentan, como los lenguajes explícitos, con librerías de funciones de nivel inferior propias del tipo de aplicación, pero aquí es el sistema quien las selecciona por si mismo en función del objetivo a alcanzar.
- En su estado actual la programación a nivel tarea puede ser tan compleja como la de nivel robot, no es de aplicación tan general y requiere de un modelo extensivo de las piezas, características, área de trabajo y el propio robot. En los programas textuales la sintaxis es próxima al lenguaje coloquial y de alto nivel, por lo que en general la ejecución de las sentencias involucra acciones muy complejas.
- A diferencia de los lenguajes de nivel robot, que actualmente son utilizados industrialmente, los sistemas a nivel tarea están aún en estados de investigación mas o menos avanzados.

Tras esta comparación con los sistemas a nivel robot veamos cuáles son las áreas previsibles de aplicación de los sistemas a nivel tarea para pasar a continuación a su taxonomía en aquellos en que la interfaz hombre-sistema es textual y aquellos en que está basada en un sistema de modelado geométrico de sólidos.

6.4.1 Consideraciones acerca de su campo de aplicación

Un robot integrado en una célula de fabricación puede estar dedicado a procesos tales como pintura o soldadura que pueden programarse adecuadamente de forma explícita, sea por guiado, sea utilizando alguno de los lenguajes a nivel robot.

Para procesos de mecanizado y ensamblaje, en cambio, son deseables técnicas de programación a nivel de tarea basadas en modelos geométricos que, entre otras cosas, permitan en tiempo de programación detectar colisiones entre mano, pinza o herramienta y el entorno dentro del que se está operando.

Mientras la mecanización necesita de un cálculo de trayectorias y del establecimiento de una estrategia de uso de herramientas, la manipulación requiere además de una definición lo más concreta posible, desde un punto de vista ordenador, de la celda de trabajo para poder operar en ella y modificarla.

En el proceso de ensamblaje puede verse cómo la creación de un nuevo objeto compuesto a partir de otros existentes inicialmente. Esto implica establecer una base de montaje y decidir cuál será la primera pieza a manipular, sobre la que se irán ensamblando las restantes piezas componentes del objeto final. Deberá determinarse el orden en que tienen que ser montadas las piezas y cuáles son las condiciones de acoplamiento entre ellas, dos a dos o en contacto múltiple.

Tendrá que tenerse en cuenta la estabilidad del conjunto, ya que no es lo mismo realizar el montaje sobre una cinta transportadora que hacerlo sobre una mesa manteniendo fija la base del ensamblaje. En el primer caso, la deposición por gravedad será más crítica que en el segundo y cualquier movimiento para el acoplamiento de dos piezas sobre una cinta será más complejo de formular dada la necesidad de tener en cuenta el movimiento de la base en función del tiempo.

Una consideración adicional es la relacionada con un posible desplazamiento de una pieza en situación crítica de contacto situada en su ubicación relativa final, entendiéndose por tal, que no volverá a ser manipulada de forma independiente. Este desplazamiento puede darse en situaciones tan diferentes como un error estático o

dinámico en la posición del brazo o durante un movimiento acomodaticio.

Existen grandes problemas que actualmente ocupan la atención de diversos grupos de investigación, remitiendo al lector a las fuentes existentes que abordan su posible solución. Entre estos problemas se citan:

- Determinación de trayectorias libres de riesgo de colisión.
- Prensado.
- Acoplamiento de piezas.

6.4.2 Sistemas de programación a nivel tarea, textuales

Como es común a todos los sistemas de programación orientados a tareas, las operaciones de ensamblado se expresan por los efectos que se desean obtener dentro del espacio de trabajo, sobre las piezas y herramientas.

En los lenguajes textuales, esta descripción se hace mediante sentencias que pertenecen a una cierta gramática. Es decir, se puede expresar directamente una operación sin necesidad de describir explícitamente todos los movimientos que el robot debe hacer para llevarla a cabo, ya que el sistema de programación será quien convierta la sentencia a nivel tarea en una secuencia de instrucciones a nivel manipulador a través de un planificador de tareas.

En estos sistemas, es el planificador quien necesita de una descripción de todos los objetos del entorno, incluido el propio robot y las herramientas, así como de la situación inicial y de la situación final que se desea alcanzar.

La planificación puede descomponerse en tres fases principales:

Modelado del entorno. - El modelo de una tarea de ensamblaje tiene que contener la información siguiente:

- Descripción geométrica de todos los objetivos y robots que intervienen.
- Parámetros físicos de todos los objetos, tales como masas e inercias.
- Descripción cinemática de todas las ligaduras.
- Descripción de características del robot: limitaciones geométricas de las articulaciones, velocidades y aceleraciones máximas, sensores de que dispone.
- Incertidumbre asociada a las distintas posiciones.

El modelo principal del entorno es la descripción geométrica de los objetos, que actualmente proviene de bases de datos de CAD, si bien en un futuro puede preverse que se cree directamente desde sistemas de visión. Estas bases de datos si bien pueden formar parte de un sistema de CAD comercial, que es lo deseable, necesitan ser adecuadamente complementadas para orientarlas hacia este tipo de procesos.

Especificación de la tarea.- Una forma de describir al planificador de tareas las operaciones de ensamblaje, es mediante una secuencia de modelos del estado del entorno correspondientes a los sucesivos pasos de la ejecución.

Un modelo del estado del entorno no es más que el conjunto de las posiciones de los objetos en un instante. Para describir una operación, el número necesario de modelos del estado en la secuencia dependerá de la capacidad del planificador y en el límite sería suficiente la descripción únicamente del estado inicial y del final.

Los planificadores de tareas necesitan mayor cantidad de información acerca de los estados intermedios y en principio deben ser capaces de producir programas a nivel robot también más detallados. La secuencia de modelos del estado en estos sistemas se hará mediante la declaración simbólica de operaciones sucesivas, lo cual requiere que los objetos y sus características puedan designarse por nombre.

Síntesis del programa a nivel robot.- El objetivo final del planificador es la síntesis

de un programa a nivel robot partiendo de la especificación de la tarea y usando el modelo del entorno. El resultado debe ser un programa integrado por órdenes de prensado, órdenes distintos de movimiento, órdenes sensoriales y comprobación de error, lo que implica resolver los problemas citados en apartado anterior.

6.4.3 Sistemas de programación a nivel tarea basados en modelado geométrico de sólidos

Los sistemas vistos, con interfaz hombre-sistema textual, puede resultarle al usuario muy pesado de utilizar, ya que es él quien tiene que determinar las relaciones espaciales deseadas entre las piezas y expresarlas mediante una cierta sintaxis. Por otra parte, si se analizan sus características, los lenguajes de nivel tarea textuales adolecen de diversas limitaciones entre las que cabe resaltar:

- Especificar la posición de las piezas a través de relaciones espaciales que si bien facilita la programación, puede ser aún más complejo, ya que algunas de ellas pueden resultar innecesarias o redundantes, mientras otras pueden ser muy difíciles de expresar.
- Los sistemas conocidos resuelven el problema coger-situar pero no abordan las numerosas operaciones de ensamblaje que requiere modificar la posición de piezas previamente colocadas.

Al ser el ensamblaje un proceso esencialmente geométrico, espacial y físico, existen líneas de investigación que abordan el desarrollo de sistemas en que las órdenes gráficas sean automáticamente descompuestas en instrucciones para el robot. Estos sistemas se conocen por asistentes de programación de robots y pretenden salvar las limitaciones anteriores citadas, al tiempo que liberan al usuario de la

necesidad de conocer una cierta sintaxis. Estructuralmente están integrados por una interfaz hombre-sistema basada en utilidades gráficas interactivas, un planificador de tareas y un sistema de modelado geométrico de sólidos.

Interfaz hombre-sistema.- Permite al usuario moverse dentro del espacio de trabajo, dirigiéndose hacia las distintas piezas y sus características.

Las piezas son visualizadas en sus posiciones actuales y opcionalmente la configuración necesaria del brazo robot para esa posición.

Planificador de tareas.- Seleccionada una operación por el usuario, el planificador mediante un módulo de inferencia, es el encargado de desglosarlas en subtareas del robot, utilizando para ello reglas de ensamblaje.

Sistema de modelado geométrico.- Es requerido por las utilidades interactivas de la interfaz y por los especialistas.

6.5 Lenguajes naturales y programación a nivel objetivo

Aquí se dará una sucinta visión de las líneas de investigación más avanzadas tendentes a lograr sistemas de programación de robots aún a más alto nivel.

6.5.1 Lenguajes naturales

El interés por incorporar este tipo de lenguajes se apoya fundamentalmente en tres consideraciones:

- Los actuales avances de la robótica, su diversificación y su difusión industrial.
- El deseo de disponer de lenguajes cada vez de más alto nivel que liberen al operador humano a través de un proceso de transferencia de inteligencia a la maquina

- La existencia en la actualidad de elementos comercializados de reconocimiento de voz que podrían ser empleados como entrada.

En trabajos de investigación recientes se han desarrollado lenguajes cuyo vocabulario dispone de algunos sustantivos, artículos, adjetivos, preposiciones, pronombres, nombres propios, conjunciones, formas verbales y adverbios. Reconocen también femeninos y plurales en sustantivos y adjetivos.

6.5.2 Programación a nivel objetivo

La programación a nivel objetivo es definida como aquella en que únicamente se especifica el objetivo final omitiendo todos los aspectos relativos a estados intermedios.

Los sistemas de programación a este nivel requieren la utilización extensiva de técnicas de inteligencia artificial, en particular de planificadores y la dificultad de su diseño radica en la generación de los planes más que en el lenguaje de descripción de los objetivos a alcanzar.

A pesar de lo dicho, en los próximos años es de prever que aparezcan sistemas a este nivel capaces de enfrentarse a tareas simples pero en áreas de trabajo en que puedan presentarse situaciones imprevistas: exploración espacial, intervención en plataformas submarinas de extracción de crudos, etc.

6.6 Presente y requisitos futuros de la programación de robots

Vistos los distintos niveles de programación de robots, veamos cual es la experiencia de su utilización, las dificultades de programación e integración que presentan actualmente y las posibles soluciones a los problemas actuales.

El análisis lo plantearemos en tres áreas: el propio robot, el entorno de fabricación y el usuario programador que no deben entenderse como independientes, sino por el contrario muy relacionadas entre sí.

6.6.1 Los sistemas de programación y el robot

Los sistemas actualmente comercializados y muchos de los desarrollados a nivel tarea son frecuentemente *dependientes del robot* para el que fueron diseñados. Esto conlleva que el programa de una aplicación no es trasladable a otro robot y que en caso de envejecimiento de éste, o se sustituye por otro idéntico o hay que reprogramar la aplicación.

Se necesitan pues sistemas de programación independientes realmente del robot, no sólo a nivel descripción de la tarea sino también al de ejecución de ésta, y ello requiere el diseño de *interfaces de software estándar* entre el robot y los programas de aplicación del usuario.

La definición de estas interfaces necesita de un convenio sobre que información hay que enviar al robot, en qué orden, y con qué protocolo de comunicación, lo cual no es fácil de hacer, pero si se logra y los fabricantes hacen uso de ellas se podrán diseñar sistemas en que las aplicaciones sean totalmente independientes del robot.

Un segundo aspecto es que *las tareas son asignadas a priori* y de forma fija a cada robot, siendo incapaces los sistemas actuales de gestionar dinámicamente una explotación como por ejemplo servir un torno y cuando éste está ocupado, realizar otras tareas. La solución tiene que llegar por el camino de la planificación inteligente.

La mayor parte de los sistemas permiten programar un único robot y de éstos algunos disponen de elementos que facilitan sincronizar varios entre sí.

Algunos sistemas contemplan la *conurrencia* y con ellos pueden programarse directamente procesos en que intervienen más de un robot, con control independiente y sincronizaciones periódicas.

6.6.2 Relación del robot con el entorno de fabricación

Actualmente los robots intervienen en procesos de fabricación perfectamente cerrados sobre sí mismos con su teclado, su pantalla, su módulo de programación por guiado activo, sus procesadores y operan totalmente aislados de otros sistemas también basados en computador que podrían estar interviniendo. Asimismo, su interacción sensorial con otros elementos de producción es muy baja.

Para invertir esta situación y lograr que los robots dejen de ejecutar completamente aislados una secuencia fija de operaciones, es necesario lograr una fuerte *integración sensorial* acompañada del *uso extensivo de conocimientos geométricos*. Estos dos requisitos son los que permitirán al robot, no sólo escoger entre diversas alternativas sino también incorporar mecanismos de detección y corrección de errores basados en la comparación del resultado obtenido en la operación con el esperado.

6.6.3 Los sistemas de programación y el usuario

El software de un robot actual suele estar formado por un conjunto de subsistemas que pueden incluir: un analizador sintáctico, un interprete que ejecuta las ordenes, un generador de trayectorias que establece el camino de la herramienta, un módulo cinemático para el cálculo de los movimientos articulares y un servosistema para el control de cada una de las articulaciones. Los subsistemas

anteriores suelen estar distribuidos en un conjunto de procesadores y no es raro que éstos a su vez estén programados en diferentes lenguajes de bajo nivel, lo que conlleva que cualquier modificación de los programas sea muy compleja.

El resultado es que el software está fuertemente ligado a la arquitectura del hardware en que se ha desarrollado y cualquier cambio de éste tiene un coste de programación prohibitivo.

A medida que el coste del hardware disminuya y se puedan incorporar procesadores más potentes y nuevas arquitecturas, la tendencia debiera ser utilizar lenguajes de más alto nivel con sintaxis estructurada, o mejor aún, lenguajes funcionales fácilmente expandibles.

Un aspecto frecuentemente olvidado en los sistemas de programación es el desarrollo y depuración de las aplicaciones y que es crucial por las especiales características que lo rodean, entre las que cabe destacar:

- Los programas en robótica tiene efectos laterales muy complejos, su tiempo de ejecución suele ser largo y no siempre es factible la reinicialización.
- La interacción en tiempo real de la información sensorial muy a menudo no puede repetirse para analizar fallos.
- Ciertos movimientos y relaciones geométricas son muy difíciles de visualizar.

De lo anterior se desprenden necesidades tales como: poder modificar los programas en línea y continuar inmediatamente; registrar las señales de los sensores y la traza del programa; implementar simuladores.

En definitiva, es necesario un *entorno de programación adecuado* y un conjunto específico de *herramientas para la depuración fuera de línea*.

6.7 Programación del *ROB-DEX*

Como el *ROB-DEX* es un prototipo original construido en nuestro medio tanto su estructura mecánica como su arquitectura electrónica, construido a base de elementos y componentes de bajo costo existentes en nuestro mercado, es por ello que se ha optado también por elaborar un sistema de programación genuino; es decir se ha elaborado un sistema software "*a medida*" para su control y operación.

La arquitectura software del *ROB-DEX* consiste en una programación de robot de "bajo nivel" para el proceso de los algoritmos de control y regulación de los motores de las articulaciones y en "alto nivel" que vienen a ser instrucciones de línea para obtener una programación con interpretación de código a nivel robot con comandos que se ejecutan en forma secuencial, con especificación de movimientos para la realización de tareas, con el fin de realizar una intercomunicación usuario-sistema.

Para realizar la programación del *ROB-DEX* se ha empleado una combinación de lenguajes de programación general que son: el *C++ (Borland C++)*, lenguaje de programación orientado a objetos y un lenguaje ensamblador (*Macroassembler*), lenguaje de programación de bajo nivel cercano al lenguaje de máquina. El lenguaje *C++* es usado para programar y procesar los algoritmos de la cinemática, planificación de trayectorias, la modelización dinámica, control del robot (programación de bajo nivel del *ROB-DEX*), también el tipo de tareas, interface con el usuario, detección de fallas, etc. (programación de alto nivel del *ROB-DEX*) y el lenguaje *ensamblador* se usa para configurar y programar la interface del sistema del controlador del robot, sincronizar e interactuar con el procesador del control (desarrollado en la sección 7.2.2 del el capítulo 7), el cual intercambia información con el computador central. Con el *C++* también se programa la monitorización de la

simulación gráfica y dinámica del entorno físico del *ROB-DEX* en tiempo real, usando técnicas de visualización y animación gráficas, y así poder realizar la interacción robot-control-entorno, evaluando las características del comportamiento del robot, el movimiento, la velocidad, la trayectoria y la ejecución de tareas.

El programa dedicado para accionar las articulaciones del *ROB-DEX* se encuentra en el procesador (8088 de Intel) usado como controlador y los programas de la modelización dinámica, cinemática, los algoritmos de control y los modelos de simulación gráfica se encuentran en el computador central. Mediante la simulación gráfica se puede controlar y visualizar al robot por ejemplo estando éste en un ambiente de trabajo dañino al hombre.

El sistema *Software* elaborado para el *ROB-DEX* se muestra en forma modular, compacta y simbólica mediante un diagrama de flujo en la Fig. (6.1), en este esquema se muestra el flujo de la información y de datos a través de las diferentes regiones de procesamiento, mostrando en cada una los subsistemas principales y las líneas importantes del flujo de información (control y datos). La arquitectura software clasifica a los subsistemas de procesamiento en una de las siguientes regiones:

- ***Interface con el operador.***- En esta región se encuentran los subsistemas encargados monitorear numérica y gráficamente el desplazamiento y velocidad de las articulaciones en función del tiempo, la simulación de la operación dinámica del robot, la visualización gráfica de la tarea que está realizando, etc. Esta región también permite el ingreso de los valores de referencia de las coordenadas por las cuales va pasar el efector final del robot, el tipo de tarea, la interpolación a usar y las instrucciones de “alto nivel” del *ROB-DEX*.

- **Proceso y control del sistema.-** Es la región más compleja e importante del sistema informático, aquí se encuentran los subsistemas encargados de procesar los algoritmos de la cinemática directa e inversa, la modelización dinámica, interpolación y generación de trayectorias, la evaluación del sistema de control empleado y las adaptaciones de entrada y salida. Esta región se interrelaciona con las regiones restantes para recibir y enviar información requerida.
- **Interfaz de entrada.-** En esta región se encuentran los subsistemas que se encargan de procesar las variables o datos que ingresan al sistema de proceso y control, que pueden ser los parámetros del robot (datos constantes), provenir de la interfaz con el operador y/o de los sensores de articulación.
- **Interfaz de salida.-** En esta región se encuentra el subsistema que procesa las variables que salen del sistema de control que son los valores de las nominales de las articulaciones y velocidades, las cuales se deben enviar como señales de mando a los motores de las articulaciones.
- **Diagnóstico o autocomprobación.** En esta región visualizan el estado del sistema total del robot, dando información acerca de los puntos no alcanzados por el efector final, los puntos de frontera de las articulaciones, los valores límites de los torques, velocidades y la visualización de fallos del sistema mediante sonido y alarmas gráficas.

Básicamente el sistema *Software* del *ROB-DEX* contiene los subprogramas o funciones de:

- Programación lógica del flujo de datos de E/S y de muestreo.
- Interconexión del computador central y procesador de control.
- Monitoreo de la interface gráfica del usuario.

- Simulación gráfica en tiempo real del robot.
- Planificación de tareas.
- Evaluación y transformación de la cinemática del robot.
- Interpolación polinómica de generar de las trayectorias.
- Evaluación de la dinámica del robot.
- Técnicas de control empleado.
- Control de los actuadores del robot.
- Algoritmos de control.
- Lectura de los sensores.
- Intérprete de instrucciones básicas de robot.
- Diagnóstico del estado del robot.

El “lenguaje” de la programación del *ROB-DEX* consiste en un sencillo intérprete de instrucciones de programación del robot, con un rango de aplicaciones limitado, su número de instrucciones textuales es limitado, la estructura de los programas es prácticamente lineal y utiliza estructuras de datos tipo objetos. El lenguaje de bajo nivel del *ROB-DEX* es un lenguaje dirigido a usuarios familiarizados con la programación orientada a objetos; mientras que el “lenguaje” a nivel robot no se necesita tener experiencia en programación.

CAPITULO VII

ARQUITECTURA ELECTRONICA DEL ROB-DEX

7.1 Introducción

En el diseño de la estructura electrónica de los robots se tiene que seleccionar circuitos adecuados de hardware. La elección depende de la estructura del robot, las características dinámicas del mecanismo del robot, de la selección de actuadores y de los requerimientos impuestos al diseño. La elección depende también de la tarea a realizar por el robot en particular. El robot tiene que asegurar el posicionamiento o seguir trayectorias de la herramienta requeridas por la tarea programada. Para llevar a cabo todos estos requerimientos se tiene que diseñar adecuadamente una arquitectura electrónica para controlar eficaz y óptimamente a las articulaciones, velocidades, fuerzas y otras variables del robot.

En la actualidad, con el avance de la tecnología en microelectrónica y el desarrollo de potentes computadores, hacen posible controlar robots con modelación dinámica altamente no lineal que requiere un análisis sumamente complejo. Hoy en día es posible construir electrónicamente controladores multiprocesadores (un procesador por articulación) de alto rendimiento para robots muy complejos con un buen número de grados de libertad, así como controlar robots que operan simultáneamente en una celda de trabajo en forma sincrónica.

7.2 Sistema electrónico

El sistema electrónico de un robot se basa en una arquitectura o estructura *Hardware*, que se encarga de procesar y transmitir las señales de referencia y de realimentación para controlar al robot manipulador. Esta arquitectura se relaciona estructuralmente con la potencia de cálculo necesaria y disponible en un sistema multiprocesador, normalmente un microprocesador por cada articulación, gobernados por un computador central que reparte y recibe sincrónicamente, entre los distintos procesadores las magnitudes de referencia, tanto de posición, velocidad así como de otras variables.

En este sistema debe residir todo el sistema operativo del robot para actuar en tiempo real. Cada procesador de articulación constituye un servomecanismo de posición, siendo la comunicación entre procesadores un cuello de botella en tiempo real, obteniéndose como contrapartida una potencia de cálculo muy considerable, realizándose dicha comunicación mediante un bus compartido, leyendo o escribiendo información en una memoria compartida o común. Para una integración más avanzada, el robot debe contar con una entrada/salida serial para hacer posible la comunicación con otros computadores externos y así controlar al robot remotamente o en lugares inaccesibles al ser humano.

Para el *ROB-DEX* se ha diseñado una arquitectura *Hardware* inteligente y programable que se basa en un microprocesador 8088 para controlar las cuatro articulaciones simultáneamente, gobernada por un computador central. Este sistema cuenta como soporte con una memoria EPROM (27256), y una memoria RAM (6264), Puertos Paralelos Programables (PPI 8255), Controlador de Interrupciones Programable (PIC 8259), Controlador de Acceso de Memoria Directa (DMA 8237),

Controlador de Tiempo Programable (PIT 8253), Convertidor Análogo Digital (ADC 0808), Convertidores Digital Analógicos (DAC 08), componentes TTL y CMOS, amplificadores, atenuadores, muestreadores, elementos para amplificar corriente, optoacopladores, oscilador local, etc. El sistema electrónico del *ROB-DEX* cuenta también con una fuente de poder regulada con diferentes salidas variables para activar al *Hardware* así como para excitar a las bobinas de los motores. Todo el sistema electrónico se encuentra externamente al computador central a excepción de la tarjeta de interfaz entre el PC y el 8088, la cual está ubicada en un slot de 8 bits del computador principal.

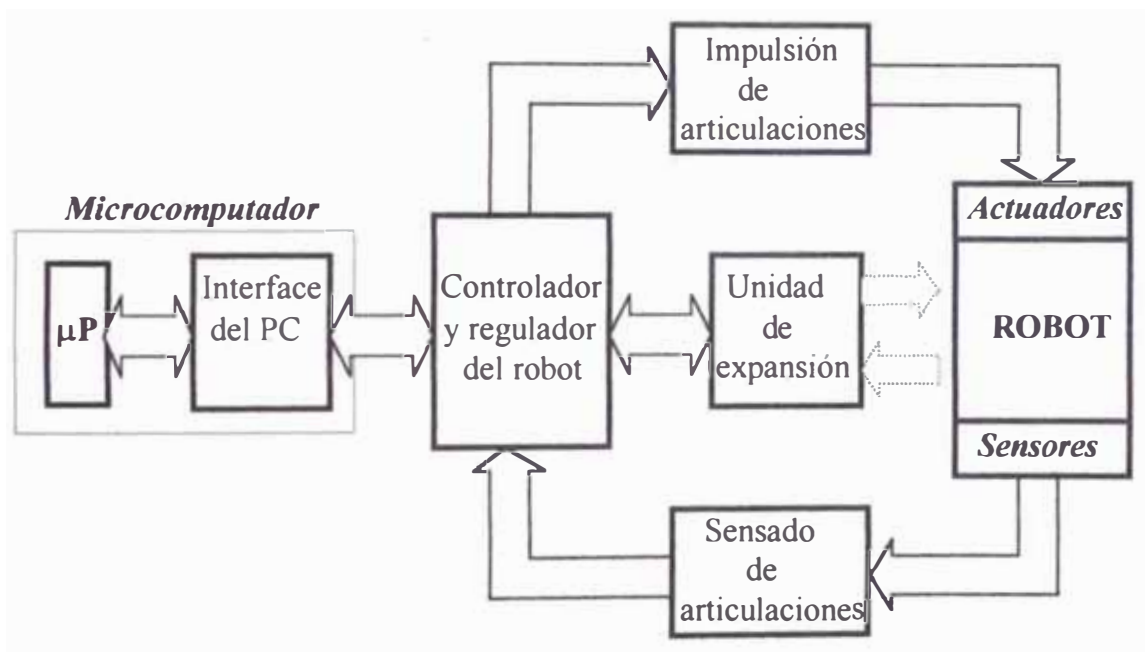


Fig. 7.1 Arquitectura electrónica del *ROB-DEX*.

La interfaz electrónica se subdivide a su vez en etapas o unidades como puede verse en forma de diagrama de bloques en la Fig. (7.1). Estas unidades son:

- Unidad de interface con el computador.
- Unidad de control y regulación del robot.

- Unidad de impulsión de los motores.
- Unidad de registro de los sensores.
- Unidad de expansión.

7.2.1 Unidad de interfaz con el computador

Esta etapa viene a ser la interfaz de *E/S* de 8 bits de datos que permite la comunicación paralela entre el *computador principal* (basado en procesadores 80X86), usado como supervisor general y el microprocesador 8088 de la etapa de control, usado como controlador del robot. En esta parte se amplifican las señales del bus de datos y se codifican las señales del bus de direcciones mediante las señales del bus de control, así como los bits de interrupciones del 80X86, es decir amplificándolos seguidamente con amplificadores de corriente y predecodificando ciertas señales para enviarlas finalmente hacia la etapa de control externa al *PC* a través de un conector de 25 pines (DB 25). Esta unidad está impresa en una tarjeta y es la que se encuentra montada en uno de los slots del computador.

7.2.2 Unidad de control y regulación del robot

Esta unidad o etapa es la parte más compleja y principal del sistema electrónico, denominada también unidad controladora de ejes del *ROB-DEX*. Esta unidad se basa en un microprocesador Intel 8088 con sus respectivos periféricos que en conjunto se comporta como el controlador digital del *ROB-DEX* para accionar a los motores de paso con señales digitales de mando, sensando la posición y velocidad de las articulaciones mediante sensores ópticos incrementales. Los motores de paso y los

sensores utilizados en el robot son fácil de controlar y registrar digitalmente. Esta unidad en forma de diagrama de bloques se puede ver detalladamente en la Fig. (7.2).

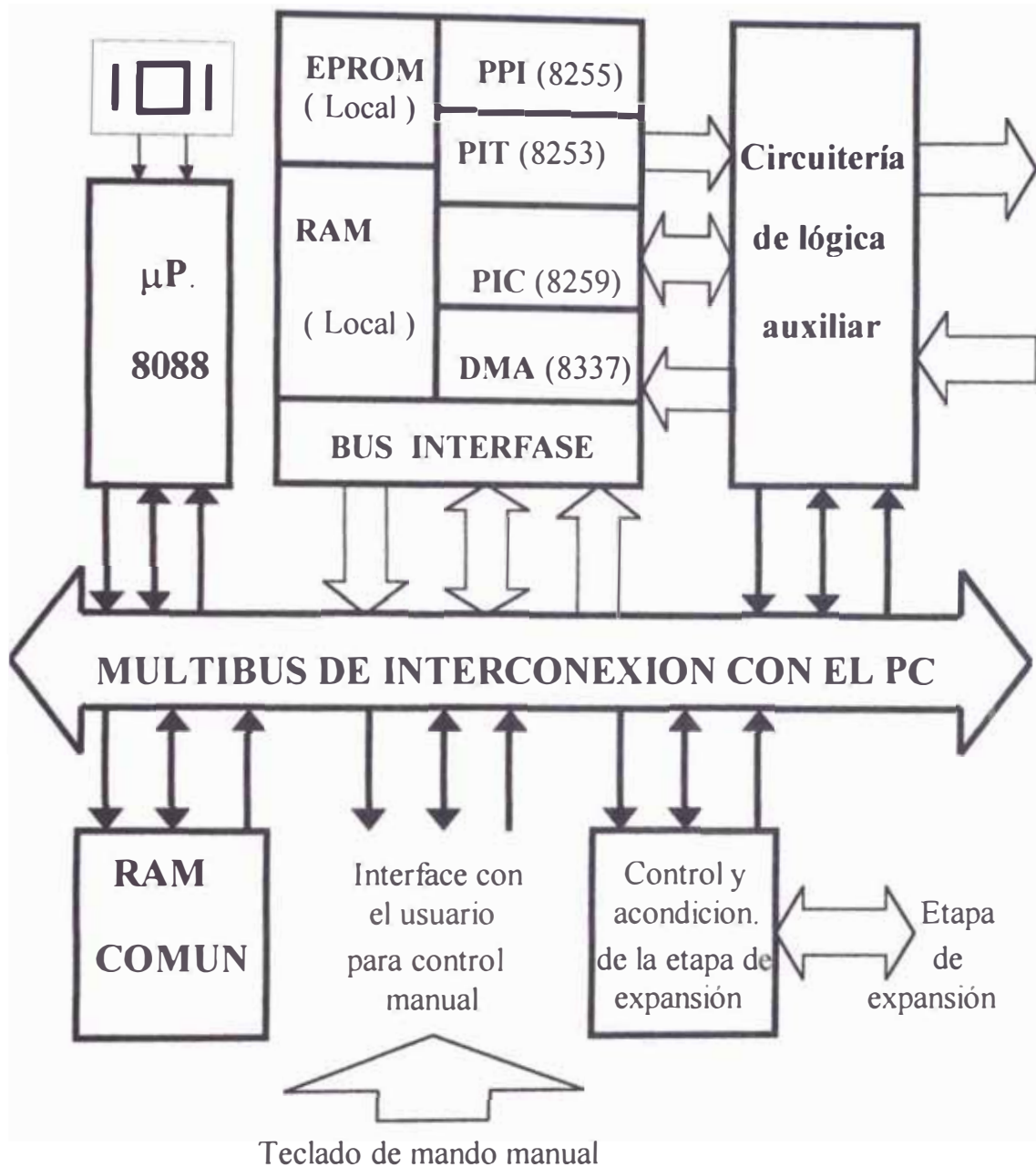


Fig. 7.2 Unidad de control y regulación de motores del *ROB-DEX*.

El 8088 después de activarse ejecuta un programa elaborado y almacenado en la EPROM local encargada de programar a los componentes programables locales (PPI, PIT, PIC, DMA) y configurar al sistema electrónico. Luego se pone a la espera de una señal del computador central; al recibir esta señal, el 8088 le envía otra señal al

PC, indicándole que se encuentra activo y dispuesto a recibir información. El computador central al recibir esta señal vuelca un programa de control del robot en la memoria común mediante el DMA del *PC*, y le indica finalmente al 8088 mediante otra señal que ya puede usar el programa, el 8088 enseguida lee la información almacenada en la *memoria común* a través del DMA local y la deposita en su RAM local; a partir de este momento interactúa en tiempo real con el *PC* intercambiando información con éste. Dependiendo de las instrucciones leídas, el 8088 puede recibir, interpretar y/o responder mensajes al computador principal cada cierto tiempo; interrumpir a éste en un tiempo preestablecido o cuando hayan fallas o límites de variables, etc., o cuando detecte falta de energía para excitar a los motores, detectar desconexiones u otras anomalías utilizando el controlador de interrupciones.

Los datos que recibe el 8088 del computador principal suelen ser valores de referencia de posición y velocidad de las articulaciones del robot. De acuerdo a estos valores el 8088 origina una frecuencia de impulsos de mando usando su propio oscilador local y los temporizadores programables y envía estos pulsos a los motores a través de los secuenciadores de la unidad de impulsión; también el 8088 recibe los valores reales de posición y velocidad de salida de los sensores a través de la unidad de sensado de articulaciones, corrigiendo en todo instante las desviaciones o errores producidos en las magnitudes de posición y velocidad angulares y enviando estas señales al computador central, en un tiempo preestablecido, para que éste los use como referencia, para la siguiente iteración, y realice la simulación gráfica en tiempo real sincronizándose a base de interrupciones.

El controlador envía las señales de mando a los actuadores, en forma simultánea a diferentes velocidades y sentidos con el fin de alcanzar los objetivos del algoritmo

principal de control procesados en el computador principal. Los valores de salida de la posición angular de cada motor, se obtiene dividiendo los valores nominales de articulación entre el ángulo de paso del motor y entre la relación de engranajes originando un número de pasos en un intervalo de tiempo de muestreo $\Delta t = 60 \text{ mseg.}$ y la velocidad se obtiene dividiendo el número de pasos del motor entre el intervalo de tiempo Δt

El controlador está diseñado para usar cualquier tipo de actuadores; es decir, es de utilidad general. Como se está usando motores paso a paso, se genera una secuencia de pulsos a una determinada frecuencia según la velocidad deseada de los motores; si se desearía implementar motores de corriente continua DC, se tendría que cambiar la unidad de impulsión de articulaciones para los motores de paso por una de impulsión de articulaciones para motores de corriente continua, y el mismo generador de pulsos de secuenciación para motores de paso de la unidad controladora se usaría para generar una frecuencia modulada por ancho de pulso, para excitar a la bobina de armadura de dichos motores, y como unidad de realimentación, se puede usar la misma de los motores de paso y así usar los mismos sensores o también se puede cambiar por otra unidad de sensado, con el fin de usar sensores de posición y velocidad angulares que emitan señales analógicas.

Las características de los motores de paso usados para el *ROB-DEX* se dan en la tabla (7.1), las cuales son usadas por el controlador de ejes para evaluar las magnitudes angulares de salida y sacar las señales adecuadas para hacer girar convenientemente a los motores; de esta manera obtener el movimiento programado.

El controlador de ejes debe estar capacitado también para recibir órdenes directamente del operario a través de un pequeño teclado para control manual, que se

usa para mover al robot manualmente o "*enseñar*" al robot o para regular la posición de los motores actuadores del sistema en forma independiente y de manera manual. Este teclado viene a ser la interface entre el operador y el *ROB-DEX*.

Por ahora el microprocesador 8088 no está implementado; tan sólo se encuentra emulado por el computador principal que hace las veces del 8088. Es decir, la programación, de los componentes programables lo realiza el *PC*, así como el flujo de la información, hacia y desde el robot, haciendo que el robot se mueva a baja velocidad, tan sólo con fines de experimentación del control del robot. Muy pronto se pondrá a punto de prueba toda la implementación; es decir, se implementará el 8088 conjuntamente con la MEMORIA COMUN, EPROM local, RAM local, el DMA local y el teclado de mando manual, así como también se deben implementar servomotores de corriente continua para obtener un desplazamiento continuo y sensores con salidas analógicas para obtener mayor resolución. Con la implementación terminada del diseño, el *ROB-DEX* se desplazará a mayor velocidad y con mayor precisión, estando disponible para someterlo en aplicaciones más rigurosas en la industria o tareas inmediatas que requieran alta velocidad y mayor precisión, o bien usarlo como prototipo para fines educativos.

7.2.3 Unidad de impulsión de las articulaciones

En esta unidad se acondicionan las señales de mando, provenientes de la unidad controladora de ejes que vienen a ser la información para originar el movimiento y la velocidad establecidos a los actuadores de articulación. Como los actuadores son motores paso a paso unipolares, la información que necesita esta unidad, son pulsos de frecuencia variable, modo de funcionamiento y sentido de movimiento. Luego de

adecuarlas convenientemente, usando una lógica secuencial, se debe suministrarles la potencia necesaria para conseguir la posición y la orientación del efector final en un punto o curva de puntos y así realizar una tarea programada.

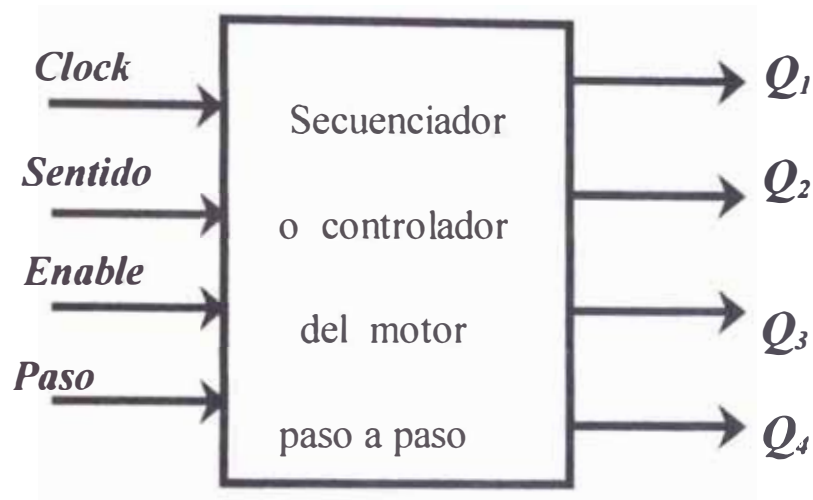


Fig. 7.3 Controlador de un motor de paso del *ROB-DEX*.

Esta unidad o etapa consta de cinco secuenciadores o controladores de los motores de paso, de los cuales cuatro sirven para controlar los motores de las articulaciones y el otro se usa para abrir y cerrar la pinza del robot. Las entradas y salidas de estos secuenciadores se pueden ver en la Fig. (7.3), donde la descripción de los bits de entrada y/o salida se describen a continuación:

- **Clock:** por este bit de entrada el secuenciador recibe una frecuencia de pulso en el flanco de bajada como señal de mando, proveniente del generador de impulsos de la unidad controladora, para hacer girar un paso por pulso a cada motor, o cada vez que se cambia la polaridad de los devanados, siendo una función de la relación de engranajes y de la secuencia aplicada. La frecuencia con la que se envían los impulsos de excitación a los devanados define la velocidad del motor.

- **Sentido:** por este bit de entrada, que proviene también de la etapa de control se indica en qué sentido debe girar el motor, dependiendo del valor que tenga, con "1" girará en sentido horario y con "0" girará en sentido antihorario.
- **Paso:** por este bit de entrada se indica si se usa paso completo o medio paso del motor. Cuando este bit toma el valor de "1" el motor realiza el paso en la secuencia adecuada que se transfiere a los devanados en forma normal y cuando es "0" el rotor gira la mitad de su recorrido normal por paso, con la ventaja de tener mayor resolución y velocidad, pero con la disminución del torque.
- **Enable:** este bit sirve para habilitar o deshabilitar al controlador y por ende al motor, ya sea por *software* o *hardware* o cuando haya una sobrecorriente en sus devanados.
- Q_1, Q_2, Q_3, Q_4 : estas cuatro señales de salida del secuenciador excitan a las bobinas de cada motor en operación unipolar, que para la inversión del flujo se devanan dos bobinas en direcciones opuestas. La secuencia de excitación de las bobinas en ambos sentidos usando paso completo se muestra en el gráfico de la Fig. (7.4).

Las señales de salida del secuenciador Q_i son salidas de nivel TTL, por lo que hay que acondicionarlas aislándolas ópticamente a través de optoacopladores T_i para seguridad del sistema electrónico, y así tener aislada la etapa de potencia de la etapa de control y luego energizar las bobinas B_i del motor de paso unipolar en lógica inversa, suministrándoles la potencia necesaria mediante transistores de potencia T_{pi} , tal como se puede ver en la Fig. (7.5).

V_{cm}, V_{c1} son voltajes variables que dependen de la potencia de cada motor, así como la resistencia R_2 que sirve para ajustar la corriente de base del transistor T_{pi} , y

por tanto, la corriente de cada bobina del motor. El diodo D_i es usado para proteger al transistor de potencia cuando se produce el flujo inverso de corriente.

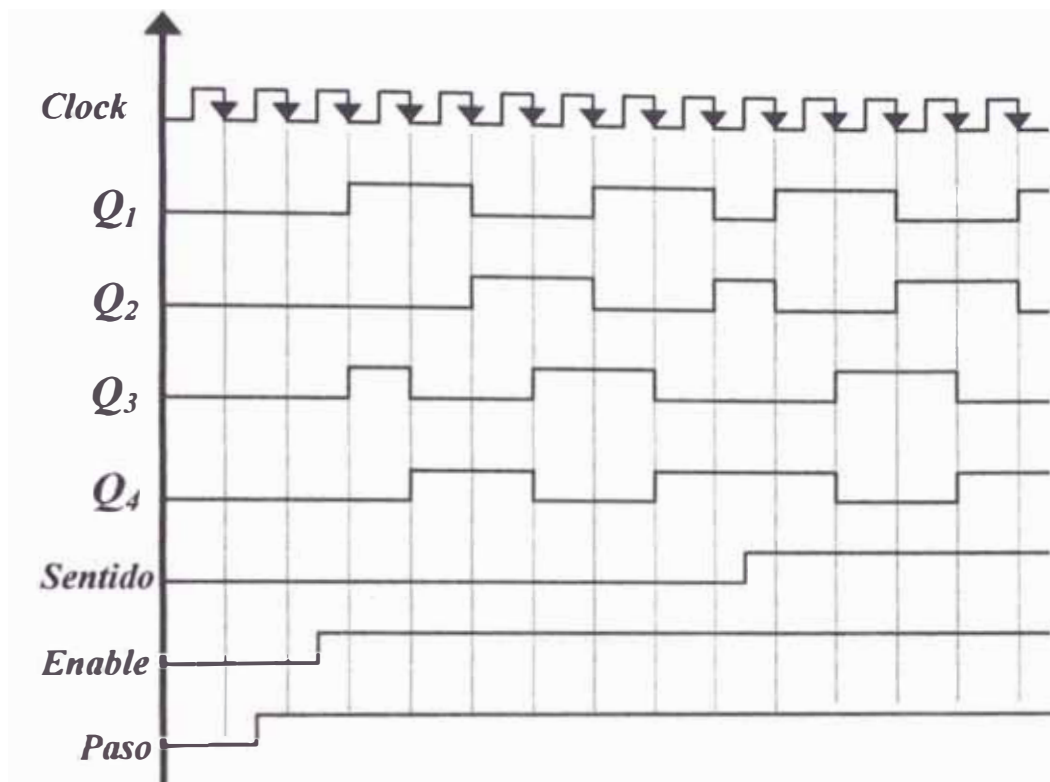


Fig. 7.4 Diagramas de tiempos del controlador del motor de paso.

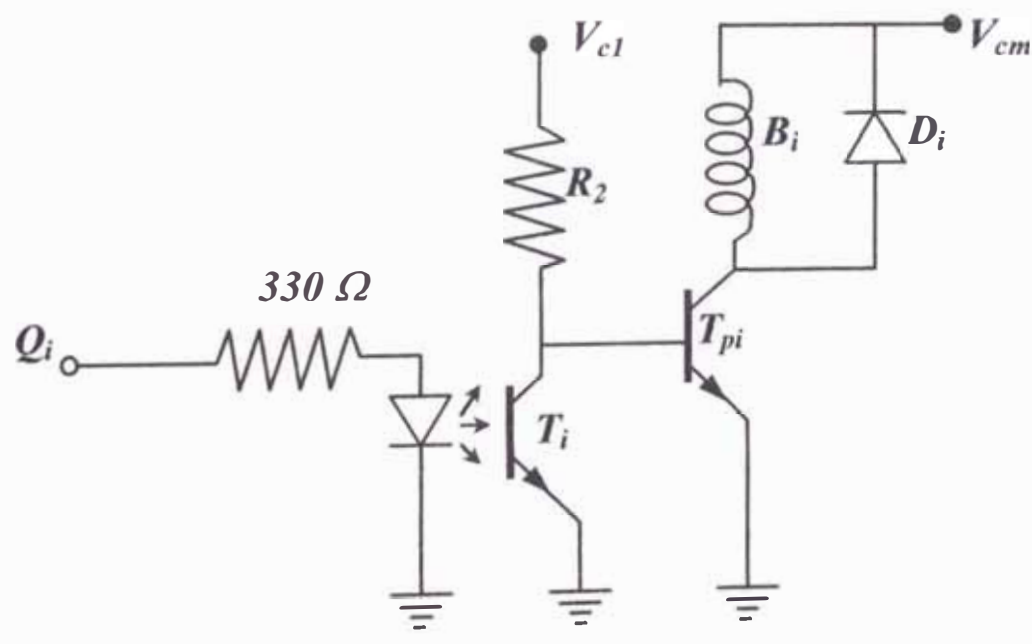


Fig. 7.5 Circuito de excitación de una bobina de un motor de paso.

El torque desarrollado por cada motor es directamente proporcional a la corriente que pasa por las bobinas e inversamente proporcional a la velocidad del motor, dada en pulsos por segundo.

En la tabla 7.1 se dan las características de los motores de paso usados para impulsar las articulaciones del *ROB-DEX*; dichas especificaciones serán usadas por esta unidad para el ajuste de corrientes de las bobinas de los motores y por el controlador de ejes para el escalado de magnitudes de posición y velocidad.

| motor | I_b/bob | V_{cm} | ángulo/paso |
|-------|------------------|----------|-------------|
| m_1 | 2.8 A | 3.8 V | 2.0 grad |
| m_2 | 1.4 A | 5.0 V | 2.0 grad |
| m_3 | 1.0 A | 5.5 V | 1.8 grad |
| m_4 | 1.8 A | 7.5 V | 7.5 grad |

Tabla 7.1 Especificaciones de los motores del *ROB-DEX*.

7.2.4 Unidad de sensado de articulaciones

En esta unidad se registran, cuantifican y digitalizan las variables que indican el estado y la situación de los elementos que componen la estructura del robot, teniendo de esta manera la información del efector final sobre su entorno físico, convirtiendo en señales eléctricas el valor de desplazamiento y velocidad de salida correspondiente a cada articulación a través de transductores o sensores.

Los sensores empleados para el *ROB-DEX* están ubicados en el eje del motor de cada articulación y se basan en codificadores ópticos incrementales que son discos ranurados en combinación de elementos optoelectrónicos que presentan garantía de

funcionamiento, un reducido tamaño, larga duración y escaso consumo de energía. Estos discos contienen dos pistas (en algunos casos, una), lo que se traduce en una mayor fiabilidad y menor coste, pero requieren una lógica adicional que permita obtener los impulsos de conteo o desconteo, según el sentido de giro y el sistema de inicialización. El sensor óptico incremental se muestra en la Fig. (7.6).

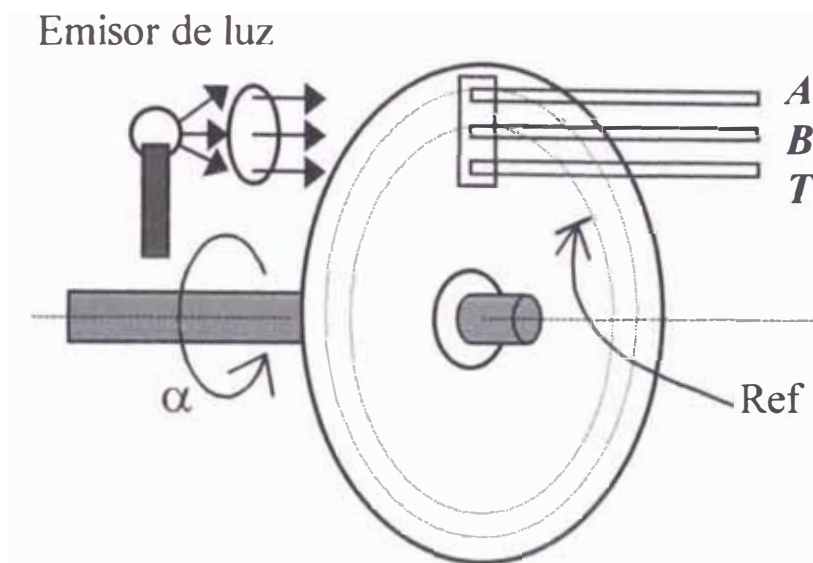


Fig. 7.6 Sensor óptico de posición angular o codificador incremental.

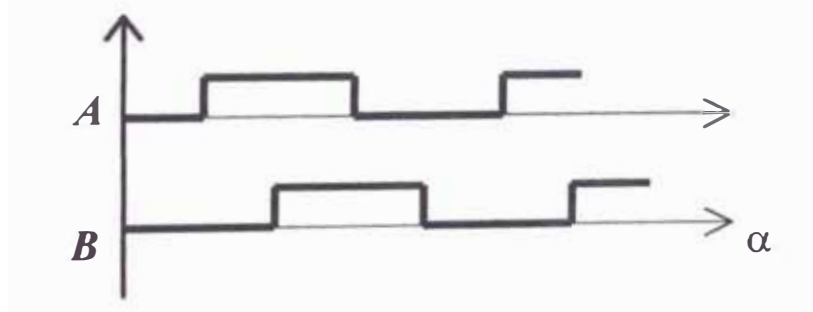


Fig. 7.7 Señales obtenidas de un codificador incremental.

La detección del sentido de giro es obtenida a partir de las señales suministradas por los fototransistores A y B (Fig. 7.6), que reciben la señal correspondiente a las dos pistas grabadas sobre el disco, cuyas marcas están desfasadas en 90° (Fig. 7.7).

Estas señales también pueden ser obtenidas por lectura de una sola pista mediante dos sensores desplazados entre sí de forma que se obtenga la señal de salida desfasada en 90° .

A partir de las señales A y B ya conformadas, pueden obtenerse cuatro impulsos de conteo por periodo como se muestra en la Fig. (7.8) mediante una lógica circuital mostrada en la Fig. (7.9).

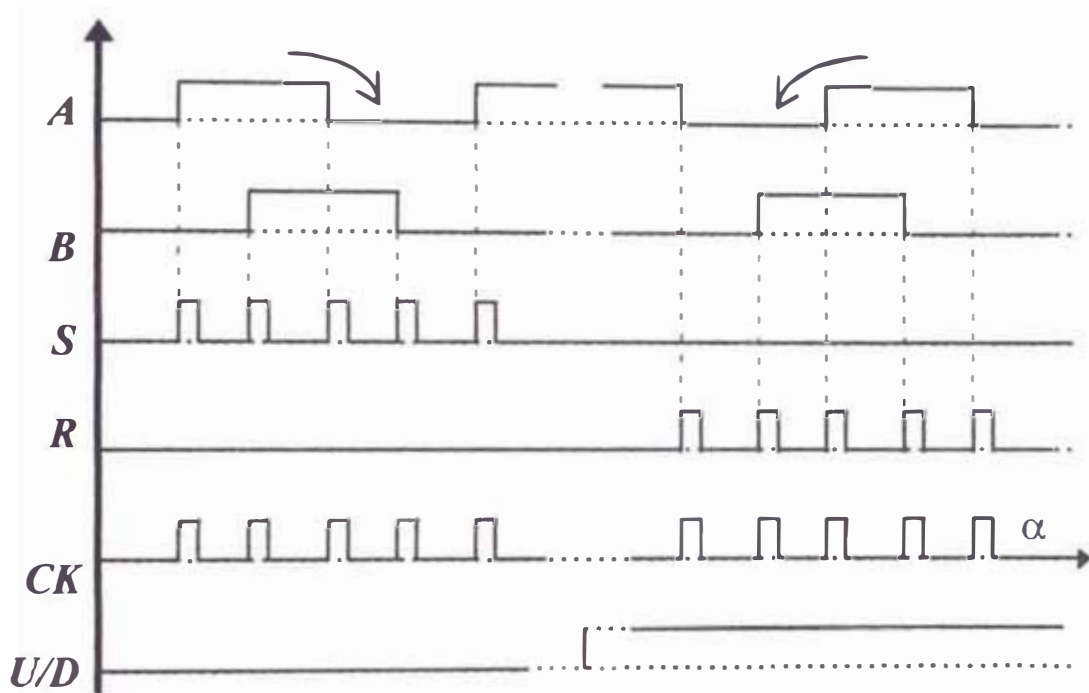


Fig. 7.8 Formas de onda de conteo y cambio de sentido.

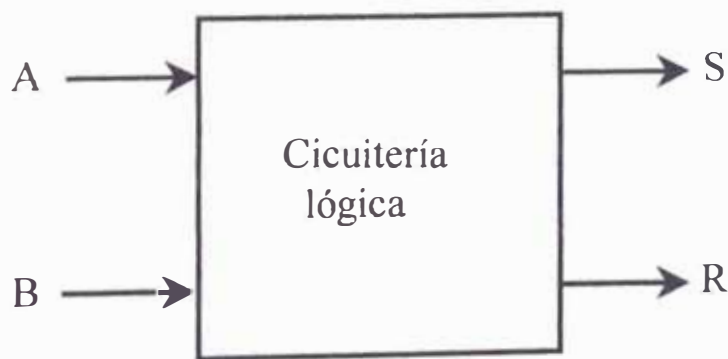


Fig. 7.9 Detector del sentido de giro de un motor.

Estos impulsos pueden ser adquiridos e integrados por la propia unidad de control o pueden utilizarse contadores exteriores Fig. (7.10) según la estructura del sistema de control utilizado.

Para aumentar la resolución a alcanzar en los codificadores ópticos incrementales, es necesario aumentar el número de franjas por revolución, lo que viene limitado por el tamaño físico de los elementos sensores.

La puesta a cero de los contadores de posición incremental es efectuada a partir de la señal de referencia que contiene el propio codificador, o a partir de detectores de final de carrera situados en la propia estructura del robot. Esta posición suele ser la de reposo al finalizar una tarea, antes de la desconexión de la unidad de control.

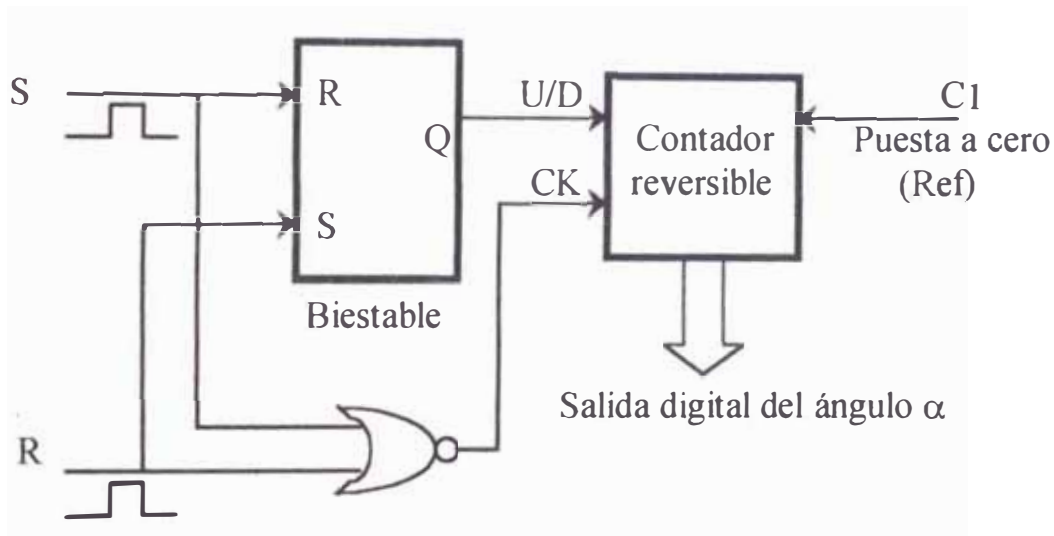


Fig. 7.10 Contador de posición angular a partir de R y S.

Las señales eléctricas de los sensores se digitalizan y se envían a la unidad de control para usarlo como retroalimentación, con el fin de regular y corregir los errores producidos, en el lazo de control del robot.

Para conseguir la máxima eficiencia en los movimientos del robot, la unidad de control, además del conocimiento de la posición de cada elemento de su estructura,

precisa efectuar también una retroalimentación en velocidad y en algunos casos también en aceleración. Ello puede ser efectuado de dos formas distintas: por cálculo a partir de la lectura periódica de la posición mediante la propia unidad de control, o si la dinámica del sistema no lo permite, utilizando transductores específicos.

Para la medición de velocidades, se emplean aún en algunos casos, los dinamos tacométricos. Estos transductores son generadores de tensión de característica tensión-velocidad muy lineal, dentro de un amplio margen de velocidades. El dinamo tacométrico tiende a no ser utilizado si se dispone de transductores de posición digitales, ya que mediante circuitos electrónicos muy simples se pueden conseguir esta información, evitándose así el coste y volumen ocupado por el dinamo tacométrico y la conversión A/D

Si el codificador es de tipo absoluto, la velocidad $v = \Delta x / \Delta t$ se puede calcular por la diferencia de posición entre intervalos de tiempo fijados por una base de tiempos.

Si el codificador es de tipo incremental, la velocidad es calculada por conteo de los impulsos obtenidos del codificador durante un tiempo Δt tal como se muestra en el gráfico de la Fig. (7.11).

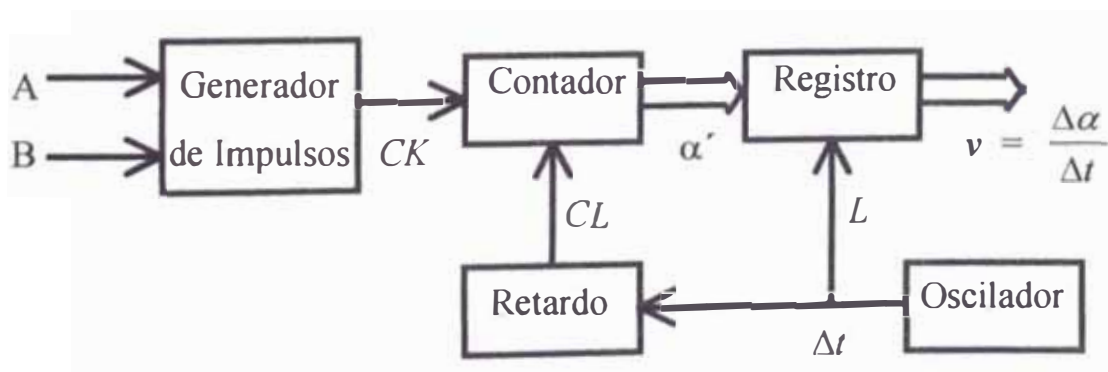


Fig. 7.11 Tacómetro digital a partir de un codificador incremental.

Cuando es necesario disponer de esta información de tipo analógico, para la retroalimentación de la unidad de mando de los actuadores, se efectúa una conversión frecuencia-tensión a partir de los impulsos obtenidos del codificador.

Para sensar la velocidad de las articulaciones del *ROB-DEX* se usa el mismo criterio de conteo de los impulsos obtenidos del codificador incremental durante un tiempo Δt , implementando un oscilador de un corto periodo de tiempo para leer en un registro la velocidad v en forma de incrementos de posición angular, este sensor es el mismo que de la Fig. (7.11).

7.2.5 Unidad de expansión del controlador

Esta unidad está diseñada para la ampliación del tratamiento de otras señales de entrada y/o salida de la unidad controladora de ejes, que pueden ser señales proporcionadas por mecanismos de detección exteriores que permitan al robot interactuar con su medio ambiente de una manera flexible, o también señales de salida del controlador para activar alarmas, señales acústicas o señales luminosas en el robot.

Los sensores de percepción externa pueden ser de fuerza, de alcance, de proximidad, de contacto, de peso, etc. De esta forma se logra aumentar la capacidad sensorial del robot para que pueda operar en contextos no restringidos. Como la mayoría de los sensores transforman magnitudes físicas a señales eléctricas analógicas, éstas se deben adecuar a valores estandarizados, ya sea amplificándolas y/o atenuándolas y/o digitalizándolas y luego enviarlas al controlador de ejes para su respectivo procesamiento.

Esta unidad está constituida fundamentalmente por convertidores análogo-digitales (ADC 0808) de 8 entradas analógicas y salidas digitales de 8 bits, de convertidores digitales-analógicos (DAC 08), amplificadores y atenuadores de voltaje, filtros, amplificadores de corriente, etc.; además cuenta con entradas y salidas digitales para señales *on-off* y con un generador de interrupciones para optimizar el tiempo computacional, dotándole al sistema de una alta interrumpibilidad.

CAPITULO VIII

APLICACIONES DEL ROB-DEX

8.1 Aplicaciones para la enseñanza de Robótica

Uno de los objetivos de la construcción del *ROB-DEX* ha sido para usarlo como prototipo de un robot que sirva para la enseñanza de los conceptos fundamentales de Robótica es decir el análisis de su modelos cinemático, de la planificación y generación de las trayectorias, guiados de robots, análisis del modelo dinámico, análisis del modelo diferencial, y las diferentes técnicas de control de un robot, así como la arquitectura mecánica, la arquitectura electrónica a base de interfaces de entrada/salida de un computador personal, interfaz robot/operador y la programación a diferentes niveles de un robot.

8.2 Aplicaciones industriales

Las aplicaciones industriales a las que se puede someter el *ROB-DEX* son diversas, limitándose a su capacidad de trabajo y dependiendo de su repetitividad, resolución, velocidad, precisión, dentro del margen de valores aceptables ya que estas características dependen de muchos factores especialmente de la arquitectura mecánica, la precisión de los motores, la velocidad de procesamiento del computador, etc. De acuerdo a su estructura física del *ROB-DEX* y a su flexibilidad que presenta, éste se puede aplicar en:

- Transporte y/o almacenamiento de bloques o piezas en cintas alimentadoras o transportadoras en una planta de producción y otras tareas de carga-descarga.
- Procesos de manutención.
- En ambientes de trabajo repetitivos, aburridos, agotadores, grasosos, radioactivos, calurosos, monótonos, enfermantes y riesgosos al hombre.
- Como ploter, en aplicaciones de textilería, cerámica, publicidad, etc.
- Con herramientas adecuadas se puede usar en soldadura por puntos, pintura, etc.
- Manipulación de máquinas herramientas.
- Inserción o ensamblamiento de pequeñas piezas.
- En sistemas de almacenamiento de anaqueles para la distribución, búsqueda, y extracción de las paletas de piezas solicitadas.

CAPITULO IX COSTOS DEL PROYECTO ROB-DEX

9.1 Introducción

El costo del proyecto del *ROB-DEX* en forma exacta es difícil de calcularlo, pero se pueden ver desde dos puntos de vista. Costos tangibles o costos reales del robot y costos intangibles. Los costos intangibles son sumamente difíciles de estimar.

Entre los costos del *ROB-DEX* se han considerados los costos de los materiales, costos de construcción, costos de diseño de Ingeniería y costos de prueba y puesta a punto. Todos los costos se han considerado en dólares USA.

9.2 Costos de materiales

| Item | Descripción | Cantidad | Costo unidad \$ | Costo total \$ |
|------|--|----------|-----------------|----------------|
| 01 | Chasis tipo CASE BABY de PC para la fijación de tarjetas impresas para el control del robot. | 01 | 60.00 | 60.00 |
| 02 | Tarjeta impresa para la interface de entrada y/o salida de información de la PC. Incluidos los componentes electrónicos y la soldadura | 01 | 40.00 | 40.00 |
| 03 | Tarjeta impresa para el control del robot incluidos los componentes electrónicos y la soldadura | 01 | 70.00 | 70.00 |

| | | | | |
|----|---|-----------|-------|--------|
| 04 | Tarjeta impresa de secuenciamiento para los motores de paso incluidos los componentes electrónicos y la soldadura | 01 | 60.00 | 60.00 |
| 05 | Tarjeta impresa de potencia para accionar los motores de paso incluidos los componentes electrónicos y la soldadura | 01 | 70.00 | 70.00 |
| 06 | 3 metros de cable de 25 pines enmallado y conectores DB-25 para la PC y caja de control. | 01 | 6.00 | 6.00 |
| 07 | Cables multifilares para la transmisión de potencia desde la caja de control hasta los motores de paso a una distancia de 5 metros. | 25 mts. | 0.50 | 12.50 |
| 08 | Material metálico para la construcción del robot: tubos cilíndricos de diversos diámetros, planchas y platinas de hierro. | 180 kilos | 1.50 | 270.00 |
| 09 | Rodajes con rodamientos cónicos (usados) de 3.5 pulgadas de diámetro interior para la primera articulación. | 02 | 10.00 | 20.00 |
| 10 | Rodajes con rodamientos cónicos (usados) de 3 pulgadas de diámetro interior para la segunda articulación. | 02 | 9.00 | 18.00 |
| 11 | Rodajes con rodamientos cónicos (usados) de 2.5 pulgadas de diámetro interior para la tercera articulación. | 02 | 8.00 | 16.00 |

| | | | | |
|----|---|----|--------|--------|
| 12 | Rodajes con rodamientos esféricos (usados) de 1 pulgada de diámetro interior para la cuarta articulación. | 02 | 4.00 | 8.00 |
| 13 | Rodaje con rodamientos esféricos (usado) de 1 pulgada de diámetro interior para la pinza del robot. | 01 | 4.00 | 4.00 |
| 14 | Juego de engranajes de nylon tipo helicoidal de dos etapas de reducción angular de razón 1/20 para la primera articulación. | 01 | 300.00 | 300.00 |
| 15 | Juego de engranajes de nylon tipo helicoidal de dos etapas de reducción angular de razón 1/9 para la segunda articulación. | 01 | 200.00 | 200.00 |
| 16 | Juego de engranajes de nylon tipo helicoidal de dos etapas de reducción angular de razón 1/2 para la tercera articulación. | 01 | 100.00 | 100.00 |
| 17 | Juego de engranajes de fierro fundido de una etapa de reducción angular de razón 4/9 para la cuarta articulación. | 01 | 10.00 | 10.00 |
| 18 | Tornillo sinfín de acero, doble entrada, 6 mm de paso, 40 cm de longitud, usado para la cuarta articulación. | 01 | 300.00 | 300.00 |
| 19 | Motor de paso unipolar (usado) de cuatro bobinas, de 2° de paso, 3.8 V, 2.8 A para activar | 01 | 25.00 | 25.00 |

| | | | | |
|----|---|----|-------|-------|
| | la primera articulación (angular). | | | |
| 20 | Motor de paso unipolar (usado) de cuatro bobinas, de 2° de paso, 5 V, 1.4 A para activar la segunda articulación (angular). | 01 | 20.00 | 20.00 |
| 21 | Motor de paso unipolar (usado) de cuatro bobinas, de 1.8° de paso, 5.5 V, 1 A para activar la tercera articulación (angular). | 01 | 18.00 | 18.00 |
| 22 | Motor de paso unipolar (usado) de cuatro bobinas, de 7.5° de paso, 7.5 V, 1.8 A para activar la cuarta articulación (prismática). | 01 | 20.00 | 20.00 |
| 23 | Motor de paso unipolar (usado) de cuatro bobinas, de 1.8° de paso, 5 V, 0.8 A para abrir y cerrar la pinza del robot. | 01 | 10.00 | 10.00 |
| 24 | Juego de pernos con sus respectivas tuercas para la sujeción de la arquitectura mecánica. | 01 | 10.00 | 10.00 |
| 25 | Pintura | 01 | 6.00 | 6.00 |

9.3 Costos de construcción

| Item | Descripción | Cantidad | Costo unidad \$ | Costo total \$ |
|------|---|----------|--------------------|-------------------|
| 01 | Construcción mecánica del <i>ROB-DEX</i> : torneado, fresado, taladrado, esmerilado, cortes, soldadura y pintado. | 01 | 1800.00 | 1800.00 |

9.4 Costos de diseño de Ingeniería

Los costos de diseño de Ingeniería no se pueden precisar con exactitud, éstos incluyen:

- Tiempo de investigación.
- Materiales de investigación: libros, revistas, fotocopias.
- Tiempo de realización del software.

9.5 Costos de prueba y puesta a punto

Los costos de prueba y puesta a punto también no se pueden precisar con exactitud, entre éstos se tiene:

- Tiempo de pruebas.
- Reemplazo de elementos o componentes fallados o de errores cometidos.
- Transporte.

CONCLUSIONES

1. El *ROB-DEX* es un producto peruano tanto en el desarrollo del *Software*, del *Hardware* y de la estructura mecánica. Tiene dos años y medio de desarrollo y se ha llegado al 80 % de implementación. Por ahora el objetivo primario es controlarlo totalmente por *PC* tanto en posición, orientación, movimiento y velocidad, y luego encargarle alguna tarea específica, fundamentalmente en aplicaciones industriales.
2. Gran parte del sistema *Hardware* está implementado en tarjetas de circuitos impresos elaboradas a base de *ORCAD*.
3. El sistema *Software* está implementado en forma mínima, sólo para fines de control del robot. Más adelante se podría mejorar la presentación del entorno windows para el usuario, la reprogramación de tareas y la facilidad de programar del operador.
4. En lo sucesivo, se cambiará al 8088 por microcontroladores para controlar a los motores DC, aumentando así su precisión, su velocidad y la versatilidad en su aplicación.
5. Con el apoyo y asesoramiento de Ingenieros Mecánicos se puede diseñar, desarrollar y mejorar la estructura mecánica, teniendo en cuenta el tipo de tareas a desarrollar, carga a soportar, volumen de trabajo, tipo de articulación, mínima frecuencia natural de vibración, alta precisión en el diseño y una optimización de los materiales empleados, aumentando así significativamente su resolución.

6. Otro de los objetivos es desarrollar más módulos de igual o diferente configuración con un equipo de ingenieros y técnicos de las diversas especialidades y con la participación de industriales para este fin que permitan implementarlos en un número de aplicaciones en la industria de nuestro país, incrementándolos conforme avance el desarrollo de sistemas avanzados y flexibles para los diversos medios de producción.
7. En la actualidad surge en nuestro medio la necesidad de implementar plantas de producción masiva flexibles a las necesidades económicas y esto se lograría con el desarrollo de robots industriales.
8. Con el trabajo realizado se trata de fomentar el desarrollo tecnológico y científico, en búsqueda de los factores y medios económicos naturales e industriales en que nuestro país lograría su desarrollo.
9. Con la Robótica se podría trabajar en los campos de la minería, metalurgia, pesquería, industria textil y manufactura.
10. En la actualidad el *ROB-DEX* está accionado por motores paso a paso, pero éstos presentan fluctuaciones instantáneas de velocidad, lo que ocasiona aceleraciones elevadas; en lo sucesivo se accionará con motores de corriente continua y sensores de percepción externa, con lo cual aumentará significativamente toda su velocidad y precisión, volviéndose más flexible, adaptable e "inteligente", creciendo así la línea de aplicaciones en su operatividad.

Los motores de paso presentan una serie de desventajas como las siguientes:

- Tienen baja velocidad, no superan las 250 R.P.M..
- Presentan un movimiento discontinuo debido a su ángulo de paso, conteniendo un error incorregible en el arco descrito.

- Normalmente los motores de paso son de baja potencia.
- El costo de este tipo de motores es elevado.

Los métodos de control para el *ROB-DEX* desarrollados en el capítulo V mostrados en las Fig. (5.13) y (5.14) se han elaborado con los siguientes fines:

- Implementar próximamente servomotores (motores DC) en el accionamiento de sus articulaciones.
- Hacer del proyecto un desarrollo de configuración y realización adaptiva con proyecciones futuras.

BIBLIOGRAFIA

- [1] Introducción a la Robótica *Oscar A. Kurisaki Uema* Seminario.
- [2] Sistemas CAD/CAM/CAE, *serie mundo electrónico*.
Editorial MARCOMBO 1986.
- [3] Robótica industrial, *Mikell P. Groover, Mitchell Weiss*.
Editorial McGraw-Hill 1989.
- [4] Curso de Robótica, *José M. Angulo Usategui, Rafael A. G.*
Editorial PARANINFO 1985.
- [5] Robótica, *Marvin Minsky* Editorial PLANETA 1985.
- [6] Analysis of mechanisms and robot manipulators, *Joseph Duffy*
Editorial Edward Arnold 1985.
- [7] Introduction to robotics, *Miomir Vukobratovic*.
Institute Mihajlo Pupin, Beograd, Yugoslavia 1989.
- [8] Fundamentals of Robotics , *Analys, Robert J. Schilling*.
Editorial PRENTICE HALL 1990.
- [9] Robotic, Control, Detection, vision and intelligence, *K. S. Fu, R. C. Gonzales*.
Editorial Mc Graw Hill 1988.
- [10] Robotic Engineering, an Integrated Approach, *R. Klafter, Chmielewski Negin*.
Editorial PRENTICE HALL 1989.
- [11] Digital Control and Estimation, *Middaleton Goodwin*.
Editorial PRENTICE HALL 1992.

- [12] Industrial Electronics and Controls, *Newman*.
Editorial PRENTICE HALL 1990.
- [13] Ingeniería de Control Moderno, *Katsuhiko Ogata*.
Editorial PRENTICE HALL 1998.
- [14] Dinámica de Lagrange *Dare A. Wells*.
Editorial McGraw-Hill 1972.
- [15] Transductores y Medidores Electrónicos, serie *Mundo Electrónico*. Editorial MARCOMBO.
- [16] Interconexión de Periféricos a Microprocesadores, serie *Mundo Electrónico*. Editorial MARCOMBO.
- [17] Microprocesadores, *José M. Angulo*. Editorial PARANINFO 1989.
- [18] Guía del programador IBM PC y PS/2, *Peter Norton*.
Editorial ANAYA 1985.
- [19] Ingeniería del Software, *Roger S. Pressman*.
Editorial McGraw-Hill 3ra. Edición 1989.
- [20] Solucionario del Programador, *Robert Jourdain*.
Editorial ANAYA 1986.
- [21] Gráficas por Computador, *Roy A. Plastock*.
Editorial McGraw-Hill 1987.
- [22] Manuales de Programación de *Borland C++*.
- [23] Manuales de Programación de *Macroassembler*.
- [24] Manual técnico de *ORCAD*.
- [25] Diversas revistas de la *IEEE*, *Mundo Electrónico*, revistas, etc.