

01173



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

DIVISIÓN DE ESTUDIOS DE POSGRADO

MODELACIÓN Y SIMULACIÓN DE UN PROBLEMA DE EVASIÓN DE OBSTÁCULOS EN EL PLANO MEDIANTE UN ROBOT DE 2 GDL APLICANDO SECUENCIAS POR COMPLEMENTOS Y ALGORITMO EOPTCE

T E S I S

QUE PARA OBTENER EL GRADO DE MAESTRO EN INGENIERÍA (ÁREA MECÁNICA) PRESENTA:
I. E. FCO. JAVIER OCHOA ESTRELLA



ASESOR: M. EN I. EUSEBIO JIMÉNEZ LÓPEZ

MÉXICO, D. F. CIUDAD UNIVERSITARIA

2004

TESIS CON FALLA DE ORIGEN



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

PRÓLOGO

Este trabajo de tesis tiene el propósito de generar conocimiento escrito para las futuras generaciones de la Maestría en Ingeniería Mecánica y campos afines, interesadas en los problemas de evasión de obstáculos.

Los robots industriales de la clase I o primera generación, tienen la desventaja de no reconocer el entorno que los rodea, requieren de un área o volumen de trabajo libre de obstáculos, pues no son capaces de detectarlos y/o evitarlos.

Muchas de las tareas industriales y de exploración, requieren de operaciones continuas ejecutadas por dispositivos robóticos. En ocasiones, el entorno de los robots es variable; es decir, se pueden presentar obstáculos tales como objetos ajenos, elementos de otras máquinas de proceso e inclusive los propios operarios.

Para proteger los robots y operarios de posibles colisiones, generalmente se aísla el área de trabajo por medio de sensores. Sin embargo, esta solución a veces no es la mas apropiada, pues si por error o accidente se activa un sensor, el robot se detiene e interrumpe su operación. Este hecho genera pérdidas de tiempo y costo y puede afectar los volúmenes de producción, máxime si los robots operan en sincronía con los dispositivos de una línea de producción.

Es necesario generar sistemas y algoritmos de detección y evasión de obstáculos que permitan a los robots reconocer su entorno, evaluar posibilidades de colisión y seguir ejecutando las tareas programadas.

En este trabajo se modela y simula un problema de evasión de obstáculos en el plano usando un robot articulado de dos grados de libertad y un obstáculo fijo. El robot ejecuta tareas usando trayectorias conocidas como de "Pick and Place" y durante su recorrido encuentra obstáculos que poseen geometría de forma circular.

El objetivo es construir los modelos cinemáticos del robot, así como determinar los criterios que permitan al robot evadir el obstáculo y seguir la tarea programada.

Se determinan criterios de evasión, donde se aplican diferentes secuencias de movimientos de los actuadores del multicuerpo, también se utiliza un algoritmo EOPTCE (Evasión de Obstáculos por Trayectoria de Contorno Expandido) para expandir la frontera del obstáculo con el propósito de que la nueva frontera pueda ser usada como trayectoria de evasión. Los modelos cinemáticos del robot se construyen usando el álgebra de los números complejos parametrizada y sistematizada en el espacio vectorial de números complejos.

Este trabajo de tesis forma parte de la línea de investigación "Modelación cinemática y dinámica" desarrollada en la sección de mecánica de la DEPMI, UNAM y por el Instituto Tecnológico Superior de Cajeme.

Finalmente, se agradece a todas aquellas personas que colaboraron para la realización de este trabajo.

RESUMEN

En este trabajo se modela y simula un problema de evasión de obstáculos en el plano, entre un robot de 2 grados de libertad y un obstáculo de geometría circular. Se analizan dos tipos de evasión:

- 1) Usando movimientos del robot y
- 2) A través de una trayectoria de contorno expandido.

En el segundo tipo de evasión se utilizó el algoritmo EOPTCE (Evasión de obstáculos por trayectoria de contorno expandido). En ambos procesos de evasión se utilizaron dos configuraciones del robot: a) codo arriba y a) codo abajo. Dichas configuraciones fueron modeladas usando el álgebra de los números complejos. La trayectoria de trabajo del robot fue del tipo pick and place y sigue un segmento rectilíneo. Fueron generados 5 criterios de evasión los cuales fueron programados y simulados con el lenguaje de programación VISUAL BASIC V6.

Palabras clave: Evasión de obstáculos, movimientos, álgebra de complejos.

ÍNDICE

PRÓLOGO		i
RESUMEN		iii
INTRODUCCIÓN		1
CAPÍTULO 1	Definición del problema, restricciones, hipótesis y axiomas.	13
	1.1 Problema de evasión de obstáculos	13
	1.2 Hipótesis y axiomas	15
	1.3 Algunas consideraciones sobre el problema de evasión de obstáculos	16
CAPÍTULO 2	Modelo de posición del robot de 2 GDL.	17
	2.1 Modelación cinemática de un robot de 2 GDL	17
	2.2 Modelación cinemática de un robot de 2 GDL utilizando la rotación ROT 1	20
	2.3 Modelación cinemática utilizando la reflexión REF 2	25
CAPÍTULO 3	Análisis y caracterización de movimientos.	31
	3.1 Definición de movimiento	31
	3.2 Tipos de movimientos completos	33
	3.3 Caracterización de movimientos uno a uno	34
	3.4 Caracterización de movimientos simultáneos	39
	3.5 Caracterización de movimientos híbridos	42
	3.6 Algunas consideraciones importantes de los movimientos completos	46
CAPÍTULO 4	Evasión mediante secuencias.	48

	4.1 Algunas consideraciones fundamentales para la planificación de trayectorias	49
	4.2 Caracterización de las posiciones relativas entre el robot y el obstáculo	58
CAPÍTULO 5	Evasión por contorno expandido.	91
	5.1 Condiciones de evasión	92
	5.2 Determinación de la trayectoria libre de colisión	94
	5.3 Condiciones de tangencia entre el obstáculo y el contorno expandido	107
CAPÍTULO 6	Programación y simulación.	117
	6.1 Programación y simulación de secuencias para el robot de 2 GDL	118
	6.2 Programación y simulación de los criterios de evasión	119
Conclusiones		152
Referencias		154
Apéndice		
	A Álgebra de complejos.	157
	B Modelos de la Cinemática Inversa.	184
	C Diagramas de flujo para los criterios de evasión.	187
	D Código de secuencias.	195

INTRODUCCIÓN

En este trabajo de tesis se pretende cumplir con los siguientes objetivos:

- Construir los modelos cinemáticos del robot.
- Caracterizar el movimiento de dos grados de libertad de los eslabones del robot.
- Establecer criterios para la planificación de una nueva trayectoria que permita al robot evadir el obstáculo y seguir su tarea encomendada.
- Validar los modelos obtenidos, desarrollando un programa de simulación que muestre: la capacidad del robot para ejecutar diferentes secuencias de movimientos, el movimiento repetitivo del órgano efector sobre la trayectoria de trabajo y la capacidad de evasión de obstáculos.

Es ampliamente conocido que la inflexibilidad de la programación manual de las tareas realizadas por un robot industrial (como el tipo PUMA), requiere que las condiciones en que se ejecutan las tareas, permanezcan invariables. Es decir, muchos de los sistemas robóticos aplicados en la industria son incapaces de reconocer su entorno para la detección de obstáculos que invadan su área de trabajo, de tal manera que para evitar accidentes que dañen a los productos, personas o al propio equipo, son detenidos totalmente para su posterior reactivación mediante la intervención de un operario.

Así se tiene que en algunas tareas industriales donde intervienen robots, como en las Celdas Flexibles de Manufactura, un problema de gran interés, lo es el de la planeación de la trayectoria a seguir por el mecanismo robótico. Básicamente este problema consiste en encontrar una secuencia de movimientos para el robot desde una cierta configuración de inicio, hasta una configuración final evitando colisiones con cualquier obstáculo del entorno.

La complejidad del problema de evasión de obstáculos es exponencial de acuerdo con el número de grados de libertad y polinomial de acuerdo al número de obstáculos en el entorno. Desafortunadamente en la industria se utilizan robots con al menos 6 grados de libertad y existen numerosos obstáculos, además la mayoría de los ambientes industriales son dinámicos, donde los obstáculos pueden estar en constante movimiento, todo lo cual convierte al problema de planificación de trayectorias en un problema muy complejo.

Para aplicaciones prácticas que permitan eficientar el funcionamiento de este tipo de máquinas, así como elevar la seguridad, es necesario, integrar al sistema robótico, la capacidad no solo de detectar la presencia de obstáculos, sino además de poder evadirlos.

Para lograr lo anterior debe existir una compatibilidad entre el plan de trabajo establecido en un sistema CAD (Computer Aided Design) con las características físicas del robot, así como las características propias de los obstáculos.

Un esquema para garantizar la ejecución de un plan de trayectorias a seguir por un robot industrial, debe tomar en cuenta las restricciones cinemáticas del mecanismo, el área de trabajo, las restricciones establecidas por los obstáculos, así como los requerimientos de las tareas a realizar, entre otras cosas.

El problema de la evasión de obstáculos se ha resuelto con diferentes técnicas, las cuales proponen una forma de representación del objeto, del robot y de su espacio de trabajo para la simulación de las tareas, por lo que un modelado apropiado evitará cálculos excesivos al realizar la detección de colisiones.

La importancia de la simulación estriba en que una representación satisfactoria de los elementos del manipulador y su entorno hacen posible generar instrucciones de control para un sistema real; es decir, si un manipulador y su entorno tienen un modelo matemático y geométrico aceptable, la planeación de movimientos podrá generarse con mayor rapidez, por otro lado la simulación de las tareas del robot permitirá probar si es factible la realización de ciertos movimientos.

Se han planteado diferentes enfoques para la solución del problema de evasión de obstáculos, considerando la modelación matemática, la representación de objetos, así como los problemas relacionados con la programación para efectos de simulación y de control en tiempo real.

Para el caso de la simulación y del control en tiempo real, es muy importante considerar el problema de la representación de los objetos, pues según sea esta,

será posible establecer una mejor planeación de los movimientos, así como lograr un control rápido y eficiente, ya que la complejidad en la representación de los objetos, implica una etapa prolongada de preproceso antes de que los algoritmos de evasión entren en acción.

Un enfoque es representar un cuerpo por medio de esferas en forma jerárquica, según se muestra en la figura 1.1. Una de las ventajas de este método es que una esfera siempre se proyecta en un círculo, sin importar la forma en que se rote, lo cual representa una ventaja en el cálculo de colisiones, pues implica resolver sistemas de ecuaciones de segundo grado [1], la idea se compone de los siguientes aspectos:

Una esfera con el mismo radio puede ser girada a lo largo del eje formando un cilindro. El eje tubular es dividido en m listas (esferas o círculos para 3D y 2D respectivamente), se conocen la longitud del objeto a ser cubierto y los puntos iniciales y finales del eje. Se consideran también los radios de las esferas. Cada posición elegida del objeto es cubierta con una esfera de tal forma que la representación final es un conjunto de esferas traslapadas a lo largo del objeto, las cuales conforman el cuerpo del objeto con un conjunto discreto de esferas.

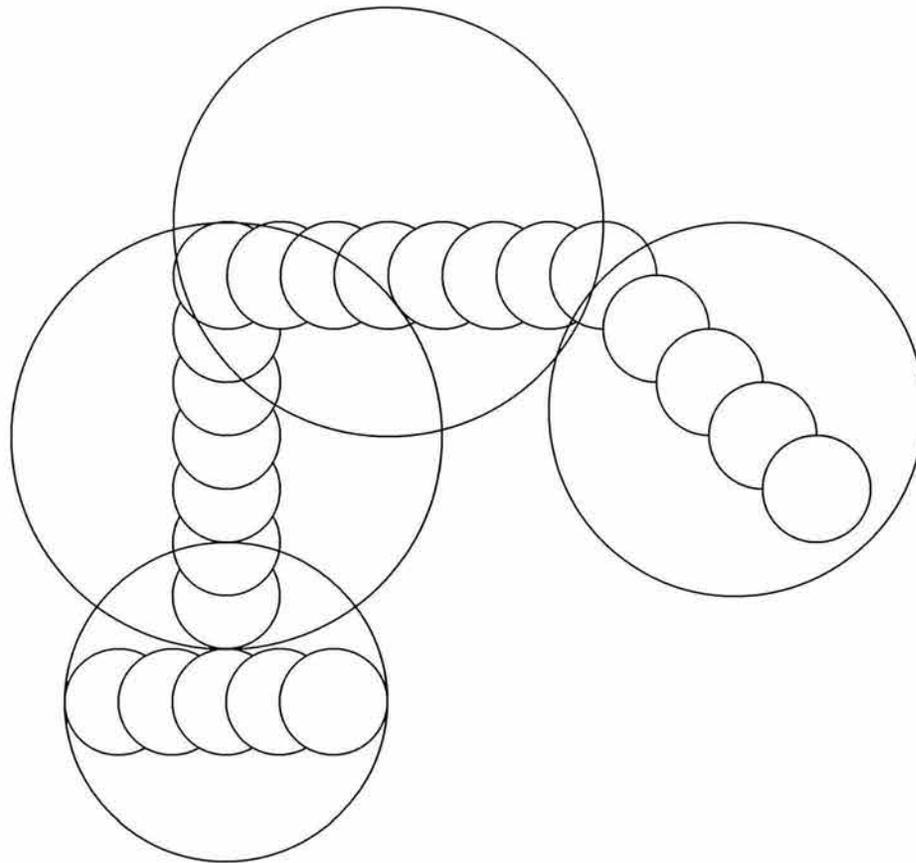


Figura 1.1 Representación del robot mediante círculos

Otra técnica para la representación de objetos y su aplicación en la detección de colisiones, se encuentra en el enfoque basado en “octrees” para 3D y “quadtrees” para 2D, ambas representaciones utilizan un sistema de representación binaria del espacio [2]. El objeto por representar es dividido recursivamente en celdas jerárquicas formando un árbol donde cada nodo representa una sección del objeto, lo anteriormente dicho, se muestra en la figura 1.2. La representación de mapas está basada en una cuadrícula que cubre el espacio de trabajo del robot, así se toma la distancia de un nodo a otro en “píxeles” para 2D y “voxels” para 3D.

Para representar un objeto por medio de este método se encierra el objeto en un cuadrado (cubo) subdividido en cuatro (ocho), que a su vez se dividen en forma recursiva dependiendo de la geometría del objeto. El cuadro o cubo inicial es la raíz del árbol y las divisiones dentro de cada uno son los nodos hijos, los nodos pueden ser de tres tipos:

- i) Blanco cuando está completamente libre.
- ii) Negro cuando está completamente ocupado.
- III) Gris cuando está parcialmente ocupado.

La idea principal en este método, es tomar las distancias mínimas y máximas (usando la distancia "Manhatan" que es la suma de las distancias en X y en Y) hacia los obstáculos en los mapas de distancias jerárquicos y compararlas con la ubicación y radio del robot para decidir si existe o no colisión. Este algoritmo requiere una etapa de preproceso la cual consiste en la construcción del árbol y el cálculo de distancias llamadas índice de separación nodal.

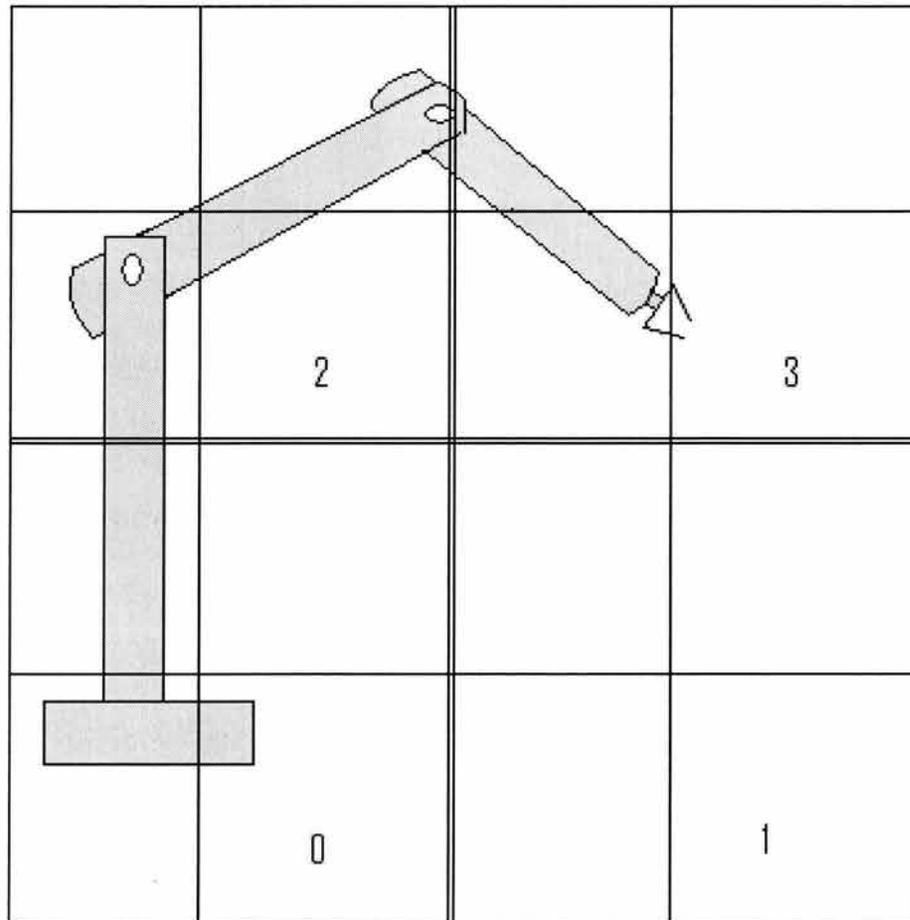


Figura 1.2 Representación de un manipulador y el cuadtree asociado.

En el trabajo propuesto en [3] se propone que para incrementar la integración entre los robots y el CAD, la planeación de las trayectorias efectuadas por el órgano efector sean establecidas antes de ejecutar algún movimiento del robot, para esto se consideran las restricciones establecidas por: la cinemática del robot, los obstáculos dentro del área de trabajo y las de las propias tareas. En este trabajo se utiliza el método denominado como "configuration space" (C-space), se

establece la modificación de la trayectoria tomando en cuenta la optimización; es decir, se trata de minimizar la desviación entre la trayectoria original y la modificada, el método anteriormente mencionado consiste en determinar previamente las diferentes posibles rutas a seguir, de acuerdo con la ubicación de los objetos y con las propiedades cinemáticas del brazo, así como a los requerimientos de las tareas, en caso de que se requiere controlar además de la posición, la orientación del órgano efector.

El trabajo desarrollado en [4] se enfoca a propiedades dinámicas, en este trabajo se simula el movimiento de un manipulador de 3 GDL y se propone un método de aproximaciones variacionales para lograr la planeación de acuerdo a un balance de energía. La ecuación del robot está dada en función de las masas de los eslabones y del órgano efector, y parten de la propiedad de que el obstáculo no es penetrable para establecer una función armónica que tiende a infinito cuando el robot se acerca al obstáculo.

En el trabajo desarrollado en [5], se presenta el método del “free Arc”, que se define como un conjunto de puntos de la configuración del robot, satisfaciendo las limitaciones de un movimiento libre de colisiones en cada tiempo de muestreo, con respecto a un sistema de coordenadas, teniendo el origen en el C-space. La definición del problema de evasión de obstáculos la hacen tomando en cuenta las limitaciones: del tiempo de movimiento, de la velocidad y las condiciones de colisiones.

En el presente trabajo de tesis, se considera un robot articulado de dos grados de libertad y un obstáculo fijo y se establecen criterios para determinar rutas de no colisión con el propósito de que el órgano efector siga una cierta trayectoria. La metodología aplicada consiste en determinar posiciones límites entre el robot y el obstáculo, a partir de las cuales es posible determinar las configuraciones y movimientos que debe realizar el robot para lograr la evasión. El alcance de este trabajo de tesis, es validar los modelos y criterios de evasión a través del desarrollo de un programa de simulación, donde se presente el movimiento del robot en un entorno numérico y gráfico.

El problema de evasión de obstáculos entre un robot de 2GDL y un obstáculo fijo, ha sido estudiado en [6,7], así como en [8]. En el primer trabajo el robot en estudio fue modelado con el álgebra de Cuaterniones [9] y el problema de evasión se solucionó usando secuencias de movimientos [10]. En el segundo trabajo, el robot se modeló usando rotaciones y reflexiones en el plano [11] y la evasión se realizó con criterios de intersección entre líneas y circunferencias. En el trabajo desarrollado en [8] el robot se modeló con una rotación y una reflexión y la evasión se realizó por un barrido sobre el obstáculo. Actualmente se desarrolló un trabajo [12] en donde se prueba la evasión del robot con un prototipo real usando el lenguaje de programación DELPHI. Finalmente, en [13] se modelaron las secuencias de movimientos de un robot paralelo planar. En este trabajo de tesis se usarán las rotaciones y reflexiones en el plano [11] para modelar y simular un problema de evasión de obstáculos en el plano caracterizado por un robot de dos grados de libertad y un obstáculo de forma circular

Este trabajo está dividido en 6 capítulos y cuatro apéndices los cuales se resumen a continuación.

En el capítulo 1 se establecen las principales restricciones a tomar en cuenta para definir las condiciones de colisión, se define el problema principal, se establece también una hipótesis fundamental y un sistema de axiomas que son utilizados para la caracterización del sistema integrado por el robot, la ruta de trabajo y las propiedades geométricas del obstáculo.

En el capítulo 2 se establece un modelo de la cinemática de posición del robot aplicando el álgebra de los números complejos, particularmente se establecen los modelos correspondientes a la posición deformada y no deformada aplicando una rotación y una reflexión [11].

En el capítulo 3 se realiza una caracterización de los movimientos que puede realizar el robot para pasar de una cierta configuración inicial a una configuración final y tocar dos puntos arbitrarios en el plano, así mismo se caracteriza el lugar geométrico descrito por el órgano efector.

En el capítulo 4 se efectúa un análisis para definir condiciones generales de colisión entre el robot y un obstáculo presente dentro del área de trabajo, se establecen criterios y secuencias de evasión, particularmente definiendo

algoritmos que implican el uso de movimientos complementarios en los actuadores.

En el capítulo 5 son analizadas las condiciones de colisión entre el obstáculo y el segundo eslabón del robot, definiendo bajo que condiciones se puede lograr una evasión a través de una trayectoria completamente paralela al contorno del obstáculo y bajo que condiciones la trayectoria de evasión debe incluir un segmento rectilíneo, además de un segmento paralelo al contorno del obstáculo.

En el capítulo 6 se establecen diagramas de flujo para la construcción de los algoritmos de evasión, así mismo se da una explicación de las funciones principales del código del programa de simulación.

En el apéndice A se establecen algunas propiedades de los números complejos.

En el apéndice B se presentan las soluciones de la cinemática inversa aplicando las operaciones de rotación 1 (ROT1) y reflexión 2 (REF2) utilizando para ello el paquete MATEMÁTICA [14].

En el apéndice C se presentan diferentes diagramas de flujo que caracterizan los 5 criterios de evasión programados.

En el apéndice D se muestra el código correspondiente al módulo de secuencias.

Al final se presentan las conclusiones y recomendaciones, así como las referencias usadas en este trabajo de tesis.

Capítulo I

Definición del problema, restricciones, hipótesis y axiomas

Introducción. En este capítulo se define el problema de evasión de obstáculos, sus restricciones fundamentales, una hipótesis y un sistema de axiomas. Se discuten algunas consideraciones de interés relacionadas con el problema de evasión de obstáculos en el plano. Se usarán los pasos generales del Método científico para plantear y solucionar el problema [15,16].

I.1 Problema de evasión de obstáculos

Se requiere modelar y simular el comportamiento cinemático de posicionamiento de un robot (R_o) de 2 grados de libertad cuando encuentra obstáculos (O_b) en el área de trabajo (A_T). El robot debe ser capaz de evadir los obstáculos localizados en posiciones específicas de A_T o en su caso, determinar si es imposible la evasión.

Las restricciones del problema planteado anteriormente son las siguientes:

- 1) El robot posee dos eslabones rígidos articulados por juntas rotacionales cada uno de ellos puede girar 360° y está empotrado en una base fija (R_1).
- 2) Las dimensiones principales de los eslabones, siendo L_1 y L_2 las longitudes del primero y segundo eslabón, satisfacen la restricción $L_1 > L_2$ (R_2).
- 3) La trayectoria (τ) ejecutada por el robot es rectilínea del tipo Pick and Place (R_3).
- 4) El área total de trabajo (A_T) es conocida (R_4).
- 5) El obstáculo posee geometría circular y se caracteriza por la siguiente relación, siendo O_b un obstáculo:

$$O_b = I_o \cup \delta_o \quad (R_5).$$

Aquí, I_o es el interior de O y δ_o su frontera (R_5).

- 6) El obstáculo permanece fijo en el área de trabajo (R_6).
- 7) El robot evade o no evade el obstáculo de acuerdo con la posición de dicho obstáculo (R_7).
- 8) El movimiento del robot es en el plano X-Y (R_8).

El problema y sus restricciones planteados anteriormente se pueden sintetizar de la manera siguiente:

“Conocido R_o , O_b , la posición de O_b en A_T y τ , encuentre CE

tal que:

$$R_o \cap O_b = \emptyset \text{ en } A_T$$

y las restricciones:

$$1) R_1 = \hat{R}_1$$

$$2) R_2 = \hat{R}_2$$

$$3) R_3 = \hat{R}_3$$

$$4) R_4 = \hat{R}_4$$

$$5) R_5 = \hat{R}_5$$

$$6) R_6 = \hat{R}_6$$

$$7) R_7 = \hat{R}_7$$

$$8) R_8 = \hat{R}_8$$

sean satisfechas ”.

Aquí, CE es un conjunto de criterios de evasión.

I.2 Hipótesis y axiomas

En esta sección se formula una hipótesis, así como un sistema de axiomas la cual permitirá solucionar el problema planteado en la sección anterior.

Hipótesis:

Si O_b es tal que: O^E con $O_b \subset O^E$ en A_T y si $R_o = T \cup E_1 \cup E_2$ con $J_1 = T \cap E_1$ y $J_2 = E_1 \cap E_2$, entonces:

$$R_o \cap O_b = \emptyset$$

sí y solo si:

existe $K: O \rightarrow O^E$ tal que:

$K(O) = O^E$ es una función de expansión.

$$J_1 \cap \delta O^E = \emptyset$$

$$J_2 \cap \delta O^E = \emptyset$$

$$O^E \subseteq R_{AT}^{ADM}$$

Aquí :

O_b es el obstáculo.

O^E es el obstáculo expandido.

E_1, E_2 son los eslabones del robot.

J_1, J_2 son las juntas del robot.

K es la función de expansión.

R_{AT}^{ADM} son regiones admisibles del área de trabajo.

El sistema de axiomas asociado con el problema son:

Axioma 1. El movimiento del robot satisface las leyes de Newton.

Axioma 2. Para todo " O_b " en A_T existe O^E .

Axioma 3. La función $K: O_b \rightarrow O^E$ es continua.

Axioma 4. Si O^E existe en A_T tal que $R_i(R_o, O^E) = 1$, entonces:

$$4.1) R_o \cap O^E = \emptyset \text{ ó}$$

$$4.2) R_o \cap O^E \neq \emptyset$$

Aquí, $R_i: R_o \times O^E \rightarrow \{1,0\}$ es una relación de impacto definido por:

$$a) R_i(R_o, O^E) = 1 \Leftrightarrow O^E \subseteq A_T$$

$$b) R_i(R_o, O^E) = 0 \Leftrightarrow O^E \notin A_T$$

1.3 Algunas consideraciones sobre el problema de evasión de obstáculos

Los modelos establecidos en el presente trabajo, solo caracterizan la cinemática de posición del robot. La velocidad, la aceleración así como las fuerzas, no son consideradas. El modelo de posición establecido permite trabajar tanto en el sistema de coordenadas de las articulaciones como en un sistema cartesiano de referencia, a través de estos modelos y los criterios de evasión determinados, se establecen los movimientos del robot para seguir la trayectoria establecida, el movimiento se define como la transición del robot entre configuraciones distintas.

Capítulo II

Modelo de posición del robot

2.1 Modelación cinemática de un robot de 2GDL

Introducción. En este capítulo se establece el modelo cinemático de posición de un robot de dos grados de libertad [7,8,10,17], utilizando el álgebra de los complejos [11] descrita en el apéndice A.

2.1.1 Caracterización del multicuerpo R_0

El multicuerpo por analizar, en esta parte, es una cadena cinemática abierta formada por dos cuerpos rígidos acoplados por medio de juntas rotacionales y su campo de acción es el plano XY. El problema consiste en construir las ecuaciones que gobiernan la cinemática de la cadena mostrada en la figura 2.1, el extremo final del segundo eslabón, está obligado, en este caso, a seguir una trayectoria específica descrita en el plano XY, con el fin de colocar una herramienta de trabajo que simule una operación de un proceso determinado [10,17].

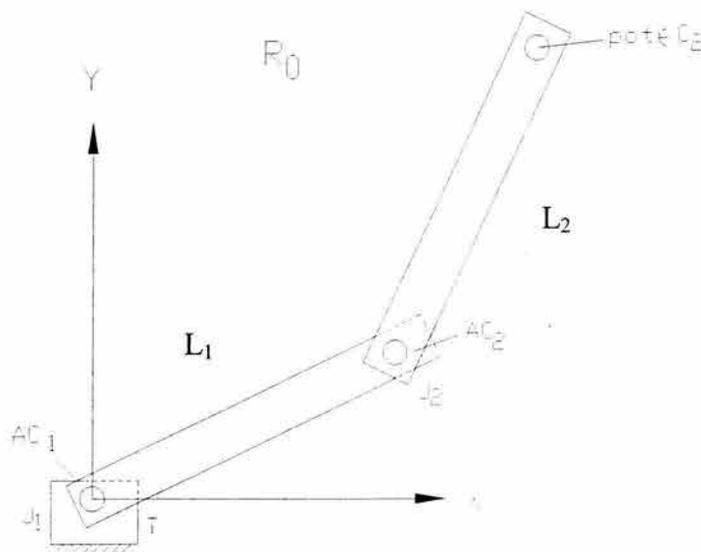


Figura 2.1 Caracterización de robot R_0

Considérese el multicuerpo (R_0) mostrado en la figura 2.1, se observa que R_0 puede ser representado por la unión siguiente:

$$R_0 = C_1 \cup C_2 \quad (1.1)$$

El multicuerpo está montado en el cuerpo T , mostrado en la figura anterior, al cual se llamará la tierra del sistema. Se denominará la junta del sistema denotada por J_1 , al conjunto de puntos dados por la intersección siguiente [10,17]:

$$J_1 = T \cap C_1 \quad (1.2)$$

En la tierra se montará un actuador, al cual será llamado AC_1 el cual se acoplará al eslabón L_1 , a través de la junta J_1 y del eje del actuador.

La junta que conecta al cuerpo L_1 con el cuerpo L_2 se define como:

$$J_2 = L_1 \cap L_2 \quad (1.3)$$

En esta parte del cuerpo se localiza un segundo actuador al cual se le llamará AC_2 y está acoplado a los cuerpos L_1 y L_2 a través de la junta J_2 .

En este trabajo el multicuerpo R_0 se considerará formado de la forma siguiente:

$$R_0 = L_2 \cup J_2 \cup AC_2 \cup L_1 \cup AC_1 \cup J_1 \cap T \quad (1.4)$$

Sobre la junta J_1 se define un sistema cartesiano inercial, también sobre el eje de L_1 se define un sistema local y sobre la junta J_2 se define el sistema de referencia del cuerpo L_2 como se muestra en la figura 2.2 [10,17]. Los sistemas de referencia asociados con los cuerpos L_1 y L_2 serán llamados en este trabajo las bases móviles y están fijas en las juntas J_1 y J_2 . Las longitudes principales de los

eslabones L_1 y L_2 serán denotadas con l_1 y l_2 . Por tanto, los vectores de posición que unen el centro de la junta J_1 con la junta J_2 y de la junta J_2 al punto $p_{ot} \in L_2$ mostrado en la figura 2.2, al cual se le llamará punto final de R_o , están definidos con respecto a las bases locales, esto es:

$$L_1 = l_1 \cdot {}^{RT1} e_1', \quad L_2 = l_2 \cdot {}^{RT1} e_1'' . \quad (1.5)$$

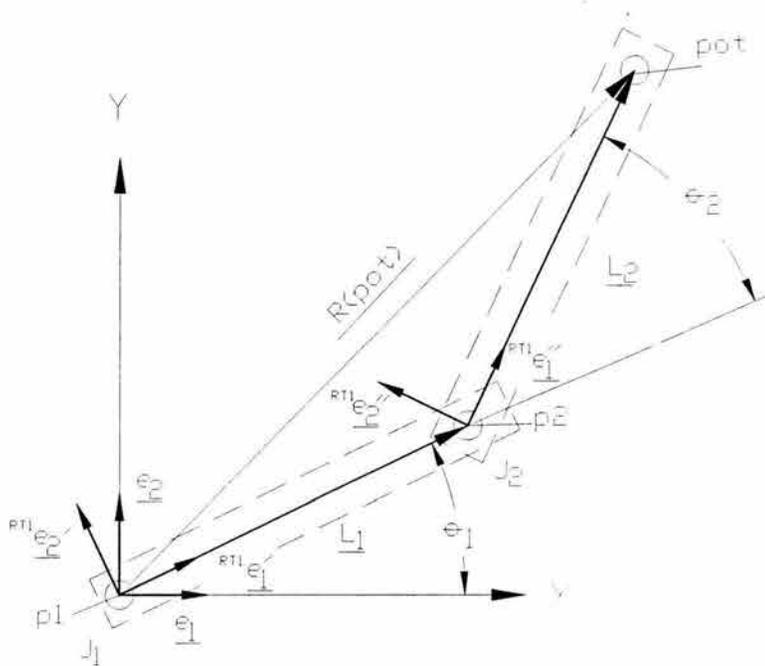


Figura 2.2 Configuración no deformada de R_o

Obsérvese que dichos vectores, $L_1, L_2 \in \mathbb{R}^2$ al fijar los puntos $p_1 \in J_1, p_2 \in J_2$ y $p_{ot} \in L_2$ pueden ser también expresados mediante la siguiente diferencia de puntos:

$$L_1 = p_2 - p_1, \quad L_2 = p_{ot} - p_2 \quad (1.6)$$

También el vector de posición de $p_{ot} \in L_2$ puede ser expresado como:

$$R(p_{ot}) = p_{ot} - p_1 \quad (1.7)$$

El vector $\underline{R}(p_{ot}) \in \mathbb{R}^2$ es representado a través de la siguiente ecuación:

$$\underline{R}(p_{ot}) = l_1 + l_2 \quad (1.8)$$

2.2 Modelación cinemática de un robot de 2GDL utilizado la rotación ROT1

En esta sección se modelarán las rotaciones del robot usando la rotación (1.2 apéndice A) y la reflexión (1.4 apéndice A) [11,17].

2.2.1 Análisis en la configuración no deformada

Considere que el multicuerpo R_o ocupa la posición mostrada en la figura 2.2. Entonces, el vector de posición de $p_{ot} \in C_2$, está dado por:

$$\underline{R}(p_{ot}) = l_1 \cdot {}^{RT1} \underline{e}'_1 + l_2 \cdot {}^{RT1} \underline{e}''_1 \quad (1.9)$$

En este trabajo se considerará que las bases móviles son rotaciones rígidas de la base inercial fija [10,17], de acuerdo con el apéndice A, tales rotaciones pueden expresarse, en el caso de complejos unitarios por:

$${}^{RT1} \underline{e}'_1 = \rho(p, \underline{e}_1) \equiv p * \underline{e}_1 \quad (1.10)$$

$${}^{RT1} \underline{e}''_1 = \rho(q, {}^{RT1} \underline{e}'_1) = \rho(q, \rho(p, \underline{e}_1)) \equiv q * p * \underline{e}_1$$

donde $\rho(p, \bullet): \mathbb{R}^2 \rightarrow \mathbb{R}^2$, $p \in \mathbb{R}^2$, es la transformación lineal [11] asociada con la rotación ROT1. Sobre la base de la expresión (1.10), la ecuación (1.9) puede ser escrita de la forma siguiente:

$$\underline{R} (p_{ot}) = l_1 \bullet \{ p * \underline{e}_1 \} + l_2 \bullet \{ q * p * \underline{e}_1 \} \quad (1.11)$$

También considerando las normas unitarias de los complejos, esto es:

$$\| p \| = \| q \| = 1 \quad (1.12)$$

se obtienen las ecuaciones que gobiernan la configuración no deformada de R_o .

Los componentes de los complejos asociados $p, q \in \mathfrak{R}^2$ son definidos en el apéndice A, esto es [11]:

$$\begin{array}{llll} p \in \mathfrak{R}^2 & p = (p_0, p_1) & p_0 \in \mathfrak{R} & p_0 = \text{Cos } \theta_1 \\ & & p_1 \in \mathfrak{R} & p_1 = \pm \text{Sen } \theta_1 \\ q \in \mathfrak{R}^2 & q = (q_0, q_1) & q_0 \in \mathfrak{R} & q_0 = \text{Cos } \theta_2 \\ & & q_1 \in \mathfrak{R} & q_1 = \pm \text{Sen } \theta_2 \end{array} \quad (1.13)$$

Aquí θ_1 y $\theta_2 \in \mathfrak{R}^2$ son los desplazamientos angulares de los eslabones medidos desde el eje X y sobre la proyección de la recta del eslabón primario.

2.2.2 Planteamiento de los problemas cinemáticos directo e inverso

En esta parte, se definirán dos problemas asociados con el multicuerpo en estudio. El primero de ellos, será el problema de la cinemática directa y el segundo corresponde a la cinemática inversa. Tales problemas están asociados a la posición mostrada en la figura 2.2, la cual será para nosotros la configuración no deformada. Considérese entonces el siguiente problema [7,8,10,17]:

2.2.2.1 Problema cinemático directo

“Dados $p=(p_0, p_1)$, $q=(q_0, q_1)$ con $\| p \| = \| q \| = 1$ y $l_1 \in \mathcal{R}^+$, $l_2 \in \mathcal{R}^+$, encuentre $R(p_{ot}) \in \mathcal{R}^2$ tal que la expresión (1.11) sea satisfecha.”

2.2.2.2 Problema cinemático inverso

“Dado $R(p_{ot}) \in \mathcal{R}^2$ y $l_1 \in \mathcal{R}^+$, $l_2 \in \mathcal{R}^+$, encuentre $p=(p_0, p_1)$, $q=(q_0, q_1)$ tal que (1.11) sea satisfecha y

$$p_0^2 + p_1^2 = 1 \quad (1.14)$$

$$q_0^2 + q_1^2 = 1$$

El problema de la cinemática inversa genera 4 ecuaciones algebraicas no lineales con 4 incógnitas del tipo polinomial [10,17], una solución obtenida computacionalmente mediante el software Mathematica V4 [14] es mostrada en el apéndice B.

2.2.3 Análisis de la configuración deformada

En esta parte se supone que el robot (R_0) ha sido movido de su posición inicial (configuración no deformada) a través de accionamientos de los actuadores AC_1 y AC_2 (ver figura 2.4). El problema ahora consiste en determinar el lugar geométrico que ocupa el extremo final del multicuerpo. La configuración así obtenida se denominará configuración deformada [7,8,10,17].

En la figura 2.4 se observa que el vector de posición $R(p_{ot})' \in \mathcal{R}^2$ correspondiente con la configuración deformada puede ser expresado de la forma siguiente:

$$R(p_{ot})' = L_1' + L_2' \quad (1.15)$$

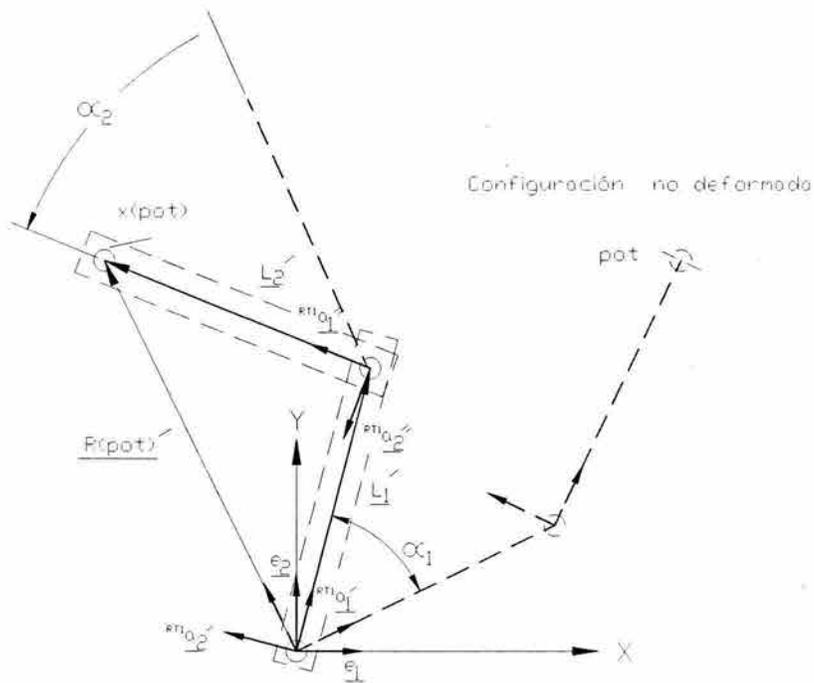


Figura 2.4 Configuración deformada de R_0

Los vectores de posición $L_1', L_2' \in \mathbb{R}^2$ representan los vectores deformados, los cuales también pueden ser escritos en términos de las bases locales, esto es:

$$1) L_1' = l_1 \cdot {}^{RT1} \underline{a}_1'$$

$$2) L_2' = l_2 \cdot {}^{RT1} \underline{a}_1''$$

Las bases ${}^{RT1} \underline{a}_1', {}^{RT1} \underline{a}_1'' \in \mathbb{R}^2$ serán en este caso, las bases móviles deformadas [7,8,10,17]. En términos de números complejos las bases deformadas se expresan de la forma siguiente:

$$, \quad {}^{RT1} \underline{a}_1' = \rho(P, {}^{RT1} \underline{e}_1') \equiv P * p * \underline{e}_1 \quad (1.16)$$

$${}^{RT1} \underline{a}_1'' = \rho(Q, \rho(P, {}^{RT1} \underline{e}_1'')) \equiv Q * P * q * p * \underline{e}_1$$

De acuerdo con la expresión (1.16), el vector de posición $\underline{R} (p_{ot})' \in \mathfrak{R}^2$ puede ser representado de la forma siguiente:

$$\underline{R} (p_{ot})' = l_1 \bullet \{ P * p * \underline{e}_1 \} + l_2 \bullet \{ Q * P * q * p * \underline{e}_1 \} \quad (1.17)$$

También considerando las normas unitarias de los complejos $P, Q \in \mathfrak{R}^2$, esto es:

$$\| P \| = \| Q \| = 1. \quad (1.18)$$

se obtienen las ecuaciones que gobiernan la configuración deformada del multicuerpo en estudio.

Al igual que en la expresión (1.13) los componentes de los complejos $P, Q \in \mathfrak{R}^2$ son [11]:

$P \in \mathfrak{R}^2$	$P = (P_0, P_1)$	$P_0 \in \mathfrak{R}$	$P_0 = \text{Cos } \alpha_1$	(1.19)
		$P_1 \in \mathfrak{R}$	$P_1 = \pm \text{Sen } \alpha_1$	
$Q \in \mathfrak{R}^2$	$Q = (Q_0, Q_1)$	$Q_0 \in \mathfrak{R}$	$Q_0 = \text{Cos } \alpha_2$	
		$Q_1 \in \mathfrak{R}$	$Q_1 = \pm \text{Sen } \alpha_2$	

2.2.4 Problema cinemático inverso en la configuración deformada

“Dado $\underline{R} (p_{ot})' \in \mathfrak{R}^2$ y $l_1 \in \mathfrak{R}^+$, $l_2 \in \mathfrak{R}^+$, $p=(p_0, p_1)$, $q=(q_0, q_1)$ con $\| p \| = \| q \| = 1$, encuentre $P=(P_0, P_1)$, $Q=(Q_0, Q_1)$ tal que la expresión (1.17) sea satisfecha y

$$P_0^2 + P_1^2 = 1 \quad (1.20)$$

$$Q_0^2 + Q_1^2 = 1$$

El problema de la cinemática inversa en la posición deformada genera 4 ecuaciones algebraicas no lineales con 4 incógnitas del tipo polinomial, igual que en el caso de la posición no deformada [17].

2.3 Modelación cinemática utilizando la reflexión REF2

En esta parte es construido el modelado cinemático del multicuerpo R_0 motivo de estudio utilizando la transformación $\rho_2(p, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ que de acuerdo al apéndice A, resulta ser una reflexión [11,17].

2.3.1 Configuración no deformada reflectada 2 de R_0

Considérese que se desea conocer la configuración del robot R_0 utilizando la transformación $\rho_2(p, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ sobre cada uno de los vectores de posición asociados con el robot motivo de nuestro estudio, como en el caso del modelado con la reflexión REF2, la configuración obtenida al aplicar dicha reflexión será llamada “configuración reflectada 2” la cual es mostrada en la figura 1.10 [17]. De igual forma que el caso estudiado anteriormente, la reflexión tendrá lugar sobre el vector de posición $\underline{R} (p_{ot}) \in \mathfrak{R}^2$ [7,8,17].

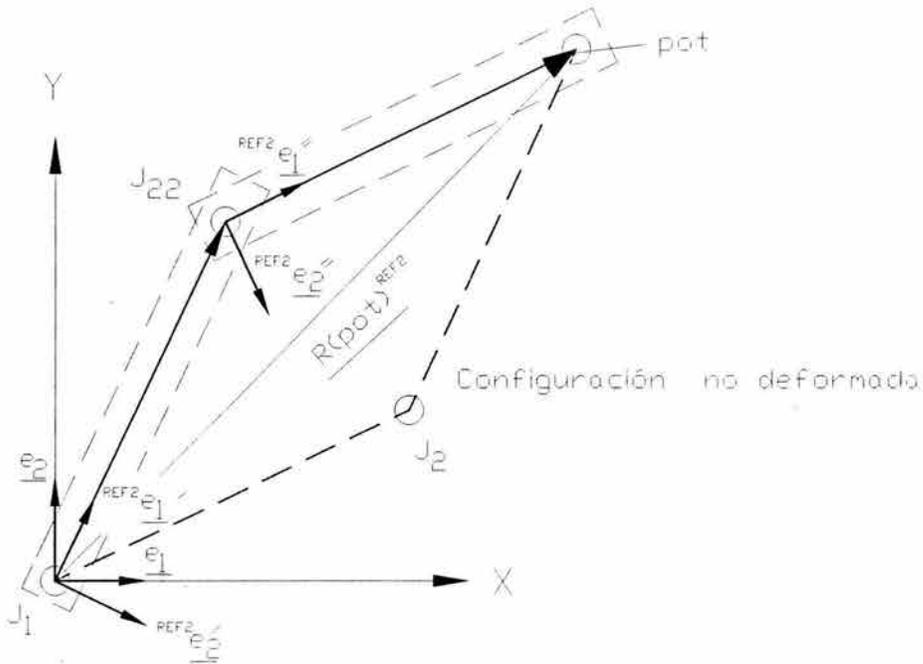


Figura 1.10 Esquematación de la configuración no deformada reflejada 2

Al observar la figura 1.10 notaremos que el vector de posición que localiza el punto terminal $p_{ot} \in C_2$ puede ser escrito en la forma siguiente:

$$\underline{R}(p_{ot})^{REF2} = l_1 \bullet \underline{e}_1'^{REF2} + l_2 \bullet \underline{e}_1''^{REF2}. \quad (1.21)$$

El vector $\underline{R}(p_{ot})^{REF2} \in \mathfrak{R}^2$ será llamado el vector reflejado bajo la transformación $\underline{R}_2(p, \bullet) : \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$, las bases reflejadas $\underline{e}_1'^{REF2}, \underline{e}_1''^{REF2} \in \mathfrak{R}^2$, de acuerdo al capítulo I, pueden expresarse de la manera siguiente [17]:

$$\underline{e}_1'^{REF2} = \rho_2(p, \underline{e}_1) \equiv p \circledast \underline{e}_1 \quad (1.22)$$

$$\underline{e}_1''^{REF2} = \rho_2(q, \underline{e}_1'^{REF2}) = \rho_2(q, \rho_2(p, \underline{e}_1)) \equiv q \circledast p \circledast \underline{e}_1$$

De acuerdo a la expresión (1.22), la ecuación (1.21) puede ser escrita de la forma siguiente:

$$\underline{R} (p_{ot})^{REF2} = l_1 \bullet \{ p \otimes \underline{e}_1 \} + l_2 \bullet \{ q \otimes p \otimes \underline{e}_1 \} \quad (1.23)$$

Los componentes de los complejos asociados $p, q \in \mathfrak{R}^2$ son definidos en el apéndice A, esto es:

$$\begin{array}{llll} p \in \mathfrak{R}^2 & p = (p_0, p_1) & p_0 \in \mathfrak{R} & p_0 = \text{Sen } \theta_1 \\ & & p_1 \in \mathfrak{R} & p_1 = \text{Cos } \theta_1 \\ q \in \mathfrak{R}^2 & q = (q_0, q_1) & q_0 \in \mathfrak{R} & q_0 = \text{Sen } \theta_2 \\ & & q_1 \in \mathfrak{R} & q_1 = \text{Cos } \theta_2 \end{array} \quad (1.24)$$

2.3.2 Problema cinemático directo

“Dados $p = (p_0, p_1)$, $q = (q_0, q_1)$ con $\| p \| = \| q \| = 1$ y $l_1 \in \mathfrak{R}^+$, $l_2 \in \mathfrak{R}^+$, encuentre $\underline{R} (p_{ot})^{REF2} \in \mathfrak{R}^2$ tal que (1.23) sea satisfecha.”

2.3.3 Problema cinemático inverso

“Dado $\underline{R} (p_{ot})^{REF2} \in \mathfrak{R}^2$ y $l_1 \in \mathfrak{R}^+$, $l_2 \in \mathfrak{R}^+$, encuentre $p = (p_0, p_1)$, $q = (q_0, q_1)$ tal que (1.23) sea satisfecha y

$$\begin{aligned} p_0^2 + p_1^2 &= 1 \\ q_0^2 + q_1^2 &= 1 \end{aligned} \quad (1.25)$$

El problema de la cinemática inversa genera 4 ecuaciones algebraicas no lineales con 4 incógnitas del tipo polinomial [17], una solución obtenida computacionalmente mediante el software Mathematica V4 [14] es mostrada en el apéndice B.

2.3.4 Configuración deformada reflectada 2 de R_0

Considérese que el multicuerpo R_0 ha sido movido de su configuración reflectada no deformada 2 a través de reflexiones sucesivas a la configuración mostrada en la figura 1.12, el vector de posición del lugar geométrico del punto $x(p_{ot}) \in C_{2t}$ es escribe de la siguiente forma [17]:

$$\underline{R}(p_{ot})^{REF2/} = l_1 \bullet \underline{a}_1^{REF2/} + l_2 \bullet \underline{a}_1^{REF2//} . \quad (1.26)$$

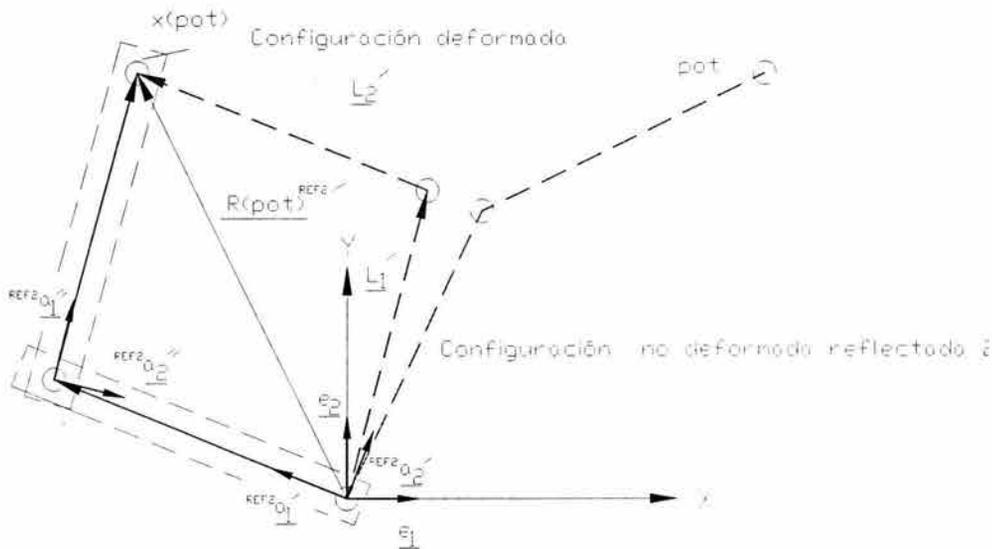


Figura 1.12 Esquematización de la configuración reflectada deformada 2

Las bases deformadas $\underline{a}_1^{REF2/}, \underline{a}_1^{REF2//} \in \mathfrak{R}^2$ están en función de números complejos, esto es [17]:

$$\underline{a}_1^{REF2/} = \rho_2(P, \underline{e}_1^{REF2/}) \equiv P \otimes p \otimes \underline{e}_1 \quad (1.27)$$

$$\underline{a}_1^{REF2//} = \rho_2(Q, \rho_2(P, \underline{e}_1^{REF2//})) \equiv Q \otimes P \otimes q \otimes p \otimes \underline{e}_1$$

De acuerdo con la expresión (1.27) el vector de posición $\underline{R} (p_{ot})^{\text{REF2/}} \in \mathfrak{R}^2$ puede ser representado de la forma siguiente:

$$\underline{R} (p_{ot})^{\text{REF2/}} = l_1 \bullet \{ P \otimes p \otimes \underline{e}_1 \} + l_2 \bullet \{ Q \otimes P \otimes q \otimes p \otimes \underline{e}_1 \} \quad (1.28)$$

Al igual que en (2.24) los componentes de los complejos P, $Q \in \mathfrak{R}^2$ son:

$P \in \mathfrak{R}^2$	$P = (P_0, P_1)$	$P_0 \in \mathfrak{R}$	$P_0 = \text{Sen } \alpha_1$	(1.29)
		$P_1 \in \mathfrak{R}$	$P_1 = \text{Cos } \alpha_1$	
$Q \in \mathfrak{R}^2$	$Q = (Q_0, Q_1)$	$Q_0 \in \mathfrak{R}$	$Q_0 = \text{Sen } \alpha_2$	
		$Q_1 \in \mathfrak{R}$	$Q_1 = \text{Cos } \alpha_2$	

2.3.5 Problema cinemático inverso en la configuración deformada reflectada

“Dado $\mathbb{R} (p_{ot})^{REF2/} \in \mathbb{R}^2$ y $l_1 \in \mathbb{R}^+$, $l_2 \in \mathbb{R}^+$, $p=(p_0, p_1)$, $q=(q_0, q_1)$ con $\| p \| = \| q \| = 1$, encuentre $P=(P_0, P_1)$, $Q=(Q_0, Q_1)$ tal que la ecuación (1.38) sea satisfecha y

$$P_0^2 + P_1^2 = 1 \tag{1.30}$$

$$Q_0^2 + Q_1^2 = 1$$

CAPÍTULO III

Caracterización de los movimientos de un robot de dos grados de libertad

Introducción

En este capítulo, se caracterizan los movimientos de un robot de 2GDL los cuales idealizan la transición de la configuración no deformada a la deformada ambas relacionadas con dicho robot. Tres tipos de movimientos son caracterizados: 1) uno a uno, 2) simultáneos y 3) híbridos [17]. La programación de tales movimientos se realiza en el paquete VISUAL BASIC V6. Los resultados obtenidos serán aplicados en los capítulos posteriores para modelar el problema de evasión de obstáculos.

3.1 Definición de movimiento

En esta sección se define el término “movimiento” el cual será utilizado para representar las transiciones entre dos configuraciones del robot R_o en estudio. Dichas transiciones se considerarán atemporales; es decir, no dependen del tiempo [17].

Sean R_o^{ND} y R_o^D las configuraciones no deformada y deformada asociadas con el robot R_o . Dichas configuraciones se muestran en la figura 3.1.

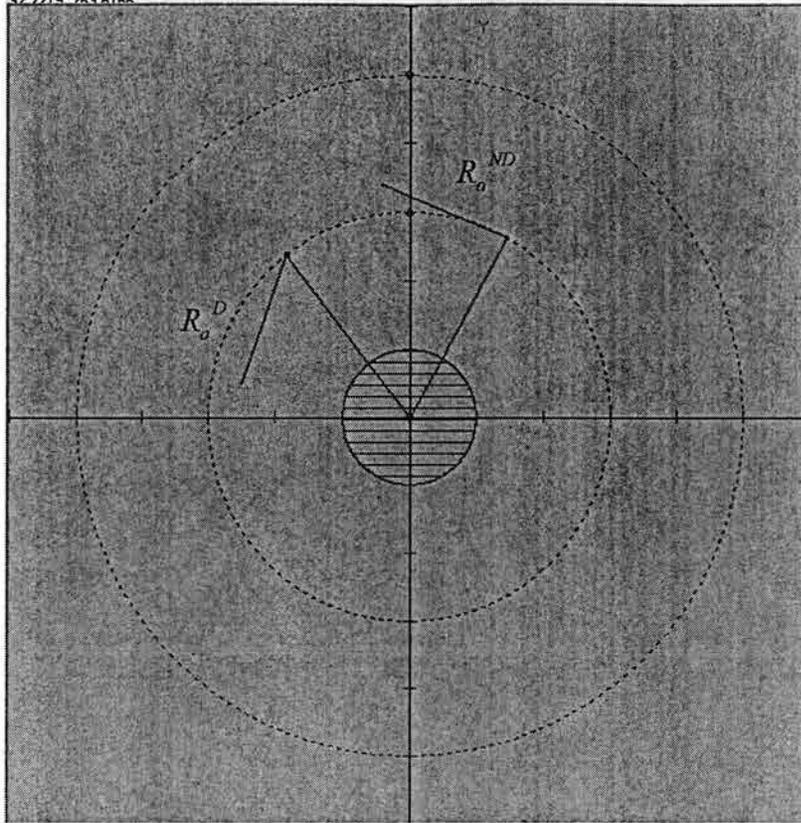


Figura 3.1 Configuraciones del robot R_0

Sea $T: R_0^{ND} \rightarrow R_0^D$. Un movimiento de R_0 se define de la manera siguiente:

“Un movimiento de R_0 es una función $T: R_0^{ND} \rightarrow R_0^D$ definida por:

$$T: [R_0^{ND}] = R_0^D$$

en donde T depende de los parámetros de rotación obtenidos del problema de la cinemática inversa. La transición $T: R_0^{ND} \rightarrow R_0^D$ es ejecutada con el problema inverso”.

Una transición o movimiento del robot se dice completa si para pasar de R_0^{ND} a R_0^D el robot fue movido usando dos parámetros de rotación asociados con sus actuadores [17].

En este trabajo de tesis se usarán movimientos completos para representar las transiciones del robot de una configuración a otra.

3.2 Tipos de movimientos completos

De acuerdo con [17,18], un robot de “n” grados de libertad puede ejecutar tres tipos de movimientos o transiciones:

- a) Uno a uno: los actuadores del robot aplican los desplazamientos angulares secuencialmente; es decir, se mueve primero un eslabón fijando el otro y, posteriormente, se mueve el eslabón restante y se fija el primero.
- b) Simultáneos: en este tipo de movimientos los actuadores aplican los desplazamientos angulares simultáneamente.
- c) Híbridos: Este tipo de movimiento se caracteriza por: 1) una activación uno a uno y 2) una activación simultánea.

Cabe señalar que los tres tipos de movimientos son completos y todos alcanzan la configuración R_o^D de R_o^{ND} . Sin embargo, las consideraciones cinemáticas relacionadas con cada movimiento son diferentes.

Supóngase que los actuadores integrados al robot R_o puedan ejecutar una revolución completa, esto es, si $P_o \in \mathfrak{R}$ es el parámetro de rotación, entonces el actuador $Ac1$ ejecutará: 1) P_o y/o P_o^* ; donde [17,18]:

$$P_o + P_o^* = 1$$

Note que $P_o(\pm)$ y $P_o^*(\pm)$ representan los casos en que un determinado actuador puede ejecutar movimientos con parámetro real $P_o \in \mathfrak{R}$ y complementario $P_o^* \in \mathfrak{R}$ con giros 1) horario (+) y 2) antihorario (-).

Bajo esta última consideración los movimientos completos se subdividen en [17,18]:

- | | | |
|-----------------|---|---|
| 1) Uno a uno: | { | Reales (P_o, Q_o)
Complementarios (P_o^*, Q_o^*)
Combinados (P_o, Q_o^*) o (P_o^*, Q_o) |
| 2) Simultáneos: | { | Reales
Complementarios
Combinados |
| 3) Híbridos: | { | Reales
Complementarios
Combinados |

Los movimientos complementarios son llamados por [17] como “reflexionados”.

3.3 Caracterización de movimientos uno a uno

En esta sección se caracterizan los movimientos completos “uno a uno” relacionados con el robot R_o motivo de estudio en este trabajo de tesis. De acuerdo con [17], el número de movimientos uno a uno (N_{MU}) relacionado con un robot de cadena abierta el cual integra eslabones activos se obtiene de la manera siguiente:

$$NMU = A + B + C$$

Aquí, $A = n!$, $B = n!$ y $C = (n!)^2$ siendo $n \in \mathbb{N}$ el grado de libertad de la cadena.

Además:

- 1) $A = n!$; es el número de movimientos uno a uno reales (N_{MUR}).
- 2) $B = n!$; es el número de movimientos uno a uno complementarios o reflexionados (N_{MURE}).
- 3) $C = (n!)$; es el número de movimientos uno a uno combinados (N_{MUC}).

Por tanto, para $n = 2$, se obtienen los siguientes resultados:

- i) $N_{MU} = 8$
- ii) $N_{MUR} = 2$
- iii) $N_{MURE} = 2$
- iv) $N_{MUC} = 4$

Así, el robot R_o motivo de estudio puede realizar 8 movimientos del tipo uno a uno. Por otro lado, de acuerdo con [17], los movimientos uno a uno se clasifican en familias y cada familia tiene un punto intermedio o, más bien dicho, una configuración intermedia generada por la primera activación de un actuador. El número de familias (N_F) y el número de puntos intermedios relacionados con un robot de cadena abierta formado de eslabones activos se obtienen de la manera siguiente:

- 1) $N_{F1} = n$
- 2) $N_{PI} = n$

La familia 1 integra los movimientos uno a uno que se caracterizan por la activación del motor M_1 en el primer submovimiento, en tanto la familia 2 contiene los movimientos uno a uno en los cuales el motor M_2 se activa en el primer submovimiento.

Los movimientos del tipo uno a uno se representan por la siguiente nomenclatura.

- 1) Movimientos reales familia y familia 2:

$$MR^{FiMj}; i = 1,2 \text{ y } j = 1,2 \text{ con } i \neq j$$

2) Movimientos reflexionados familia 1 y familia 2:

$$MRE^{FiMj}; i = 1,2 \text{ y } j = 1,2 \text{ con } i \neq j$$

3) Movimientos combinados de familia 1 y la familia 2:

$$MC_{Mk}^{FiMj}; i = 1,2 \text{ y } j = 1,2 \text{ con } i \neq j, k = 1 \text{ o } 2$$

Por ejemplo:

- i) MR^{FiM2} : es un movimiento real de la familia 1 con M2 activado en segundo término.
- ii) MRE^{F2M1} : es un movimiento complementario o reflexionado de la familia 2 con M1 activado en segundo término.
- iii) MC_{M2}^{F2M1} : es un movimiento combinado de la familia 2 con M1 activado en segundo término y M2 efectúa un movimiento complementario.

Los movimientos uno a uno se representan por medio de tablas en código binario; esto es:

MR ^{F₁M₂}		
T \ M	M ₁	M ₂
T ₁	1	0
T ₂	0	1

a)

MRE ^{F₂M₁}		
T \ M	M ₁	M ₂
T ₁	-1	0
T ₂	0	-1

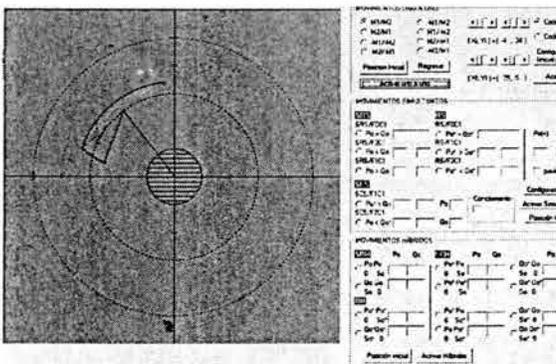
b)

MC ^{F₂M₁} M ₂		
T \ M	M ₁	M ₂
T ₁	0	-1
T ₂	1	0

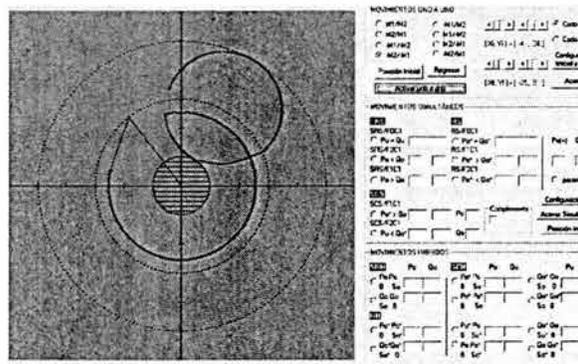
c)

Tablas 3.1 representación de movimientos uno a uno: a) reales, b) complementarios y c) combinados

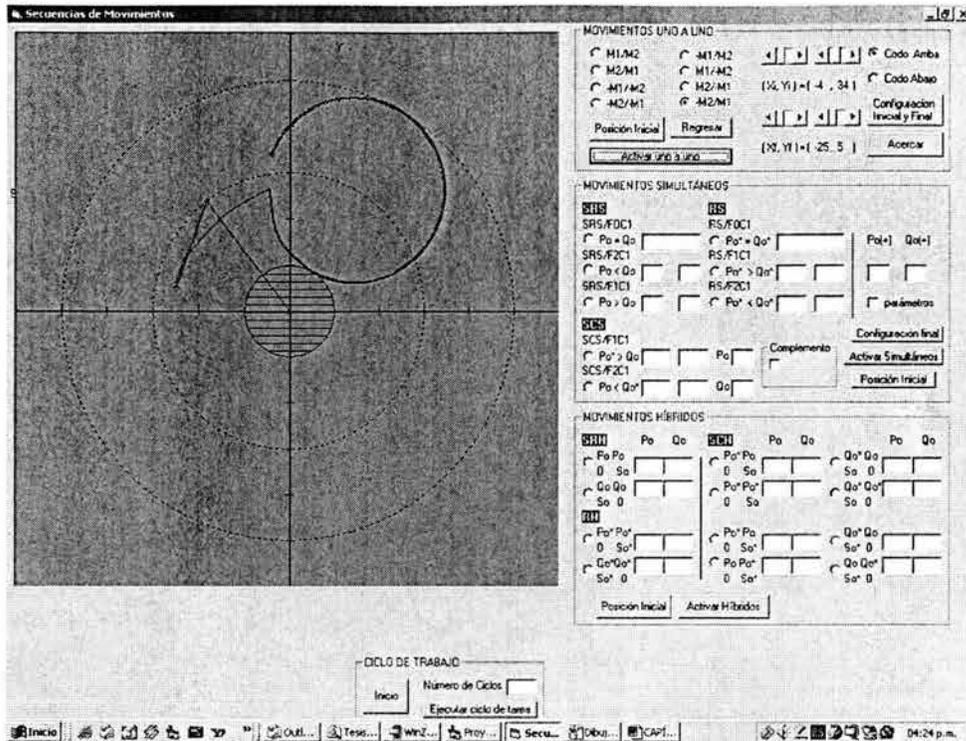
La figura siguiente muestra dichos movimientos programados en VISUAL BASIC V6.



a)



b)



c)

Figura 3.2 Salidas gráficas de movimientos: a) real, b) complementario y c) combinado

Para finalizar esta sección es importante señalar dos aspectos fundamentales de los movimientos uno a uno, estos son:

- 1) El recorrido del órgano terminal para cada movimiento uno a uno es diferente.
- 2) Los puntos que integran los eslabones del robot se desplazan en forma diferente en cada movimiento.

Cada familia integra cuatro movimientos simultáneos, es decir:

- 1) Un movimiento real.
- 2) Un movimiento reflexionado.
- 3) Dos movimientos combinados.

Es importante señalar que el hecho de que existan tres familias implica tres tipos de configuraciones deformadas relacionadas con el robot, una para cada familia, puesto que las relaciones $P_o = Q_o$, $P_o > Q_o$ y $P_o < Q_o$ implican tres configuraciones deformadas diferentes.

Por otro lado, para el caso de los movimientos simultáneos existe un número finito de configuraciones intermedias, dicho número depende del número de discretizaciones asociadas con el intervalo de tiempo sobre el cual se ejecuta el movimiento.

3.4 Caracterización de los movimientos simultáneos

En esta sección se caracterizan los movimientos del tipo simultáneos relacionados con el robot R_o motivo de estudio. En estos movimientos los actuadores que componen el robot, se activan simultáneamente y desplazan angularmente los eslabones en un mismo intervalo de tiempo.

De acuerdo con [17], los movimientos simultáneos se clasifican en tres familias:

- 1) F_o ; es la familia de movimientos simultáneos en el cual $P_o = Q_o \Leftrightarrow \alpha_1 = \alpha_2$.
- 2) F_1 ; es la familia de movimientos simultáneos en el cual $P_o > Q_o \Leftrightarrow \alpha_1 > \alpha_2$.
- 3) F_2 ; es la familia de movimientos simultáneos en el cual $P_o < Q_o \Leftrightarrow \alpha_1 < \alpha_2$.

Considere que $P_o > Q_o$, la familia F_1 integra los siguientes movimientos simultáneos:

- 1) $MRS_{P_o > Q_o}^{F_1}$; movimiento real simultáneo de la familia F_1 .
- 2) $MRES_{P_o > Q_o}^{F_1}$; movimiento complementario o reflexionado de la familia F_1 .

3) $MCS_{P_0, Q_0}^{F_1}$; movimiento combinado simultáneo de la familia F_1 con

$$M_2 \rightarrow Q_0^*$$

4) $MCS_{P_0^*, Q_0}^{F_1}$; movimiento combinado simultáneo de la familia F_1 con

$$M_1 \rightarrow P_0^*$$

Las tablas de código binario asociadas con los movimientos mencionados anteriormente son:

$MRS_{P_0 > Q_0}^{F_1}$		
T \ M	M_1	M_2
T_1, T_2	1	1

a)

$MRES_{P_0 > Q_0}^{F_1}$		
T \ M	M_1	M_2
T_1, T_2	-1	-1

b)

$MCS_{P_0, Q_0}^{F_1}$		
T \ M	M_1	M_2
T_1, T_2	1	-1

c)

$MCS_{P_0^*, Q_0}^{F_1}$		
T \ M	M_1	M_2
T_1, T_2	-1	1

d)

Tabla 3.2 Representación de los movimientos simultáneos de la familia 1: a) real, b) complementarios, c) y d) combinados

3.5 Caracterización de los movimientos híbridos

En esta sección se caracterizan los movimientos híbridos relacionados con el robot R_o motivo de estudio en este trabajo de tesis. Este tipo de movimiento combina tanto los uno a uno como los simultáneos.

De acuerdo con [17,18], un robot de 2GDL puede alcanzar la configuración deformada (R_o^D) de la no deformada (R_o^{ND}) usando movimientos especiales llamados "híbridos". Dichos movimientos se caracterizan por:

- 1) Dos submovimientos para alcanzar R_o^D de R_o^{ND} .
- 2) Un submovimiento es simultáneo; es decir, M_1 y M_2 se activan con P_o los dos o Q_o los dos.
- 3) El submovimiento simultáneo se da en la primera activación.
- 4) Un submovimiento uno a uno de corrección angular ya sea en M_1 o M_2 .

Existen dos familias de movimientos híbridos:

- 1) Familia F_1 : Familia que integra los movimientos híbridos en la cual M_1 corrige el desplazamiento angular.
- 2) Familia F_2 : Familia que integra los movimientos híbridos en la cual M_2 corrige el desplazamiento angular.

Cada familia integra los siguientes movimientos:

- 1) Un real.
- 2) Un complementario.
- 3) Tres combinados simples.
- 4) Tres combinados dobles.

Un movimiento real híbrido es el siguiente:

$$MRH^{F_1}$$

El código asociado con MRH^{F_1} es:

MRH^{F_1}		
T \ M	M_1	M_2
T_1	1	1
T_2	1	0

Tabla 3.3 Código de un movimiento real híbrido familia 1

La figura siguiente muestra la simulación del movimiento descrito anteriormente.

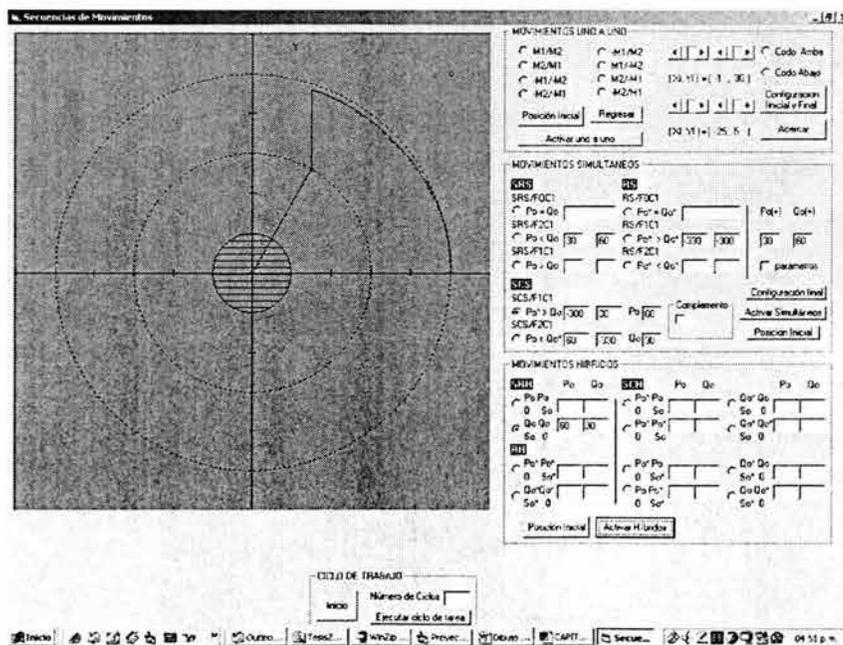


Figura 3.4 Salida gráfica del movimiento MRH^{F_1}

Las siguientes tablas muestran los códigos de los movimientos:

- 1) $MREH^{F_1}$: movimiento reflexionado híbrido de la familia 1.
- 2) $MCH^{F_1}_{M_1, T_1}$: movimiento combinado híbrido simple con submovimiento complementario en M_1 .
- 3) $MCH^{F_1}_{M_{1,2}, T_{1,1}}$: movimiento combinado híbrido doble con submovimientos en M_1 y M_2 .

$MREH^{F_1}$		
T \ M	M_1	M_2
T_1	-1	-1
T_2	-1	0

a)

$MCH^{F_1}_{M_1, T_1}$		
T \ M	M_1	M_2
T_1	-1	1
T_2	1	0

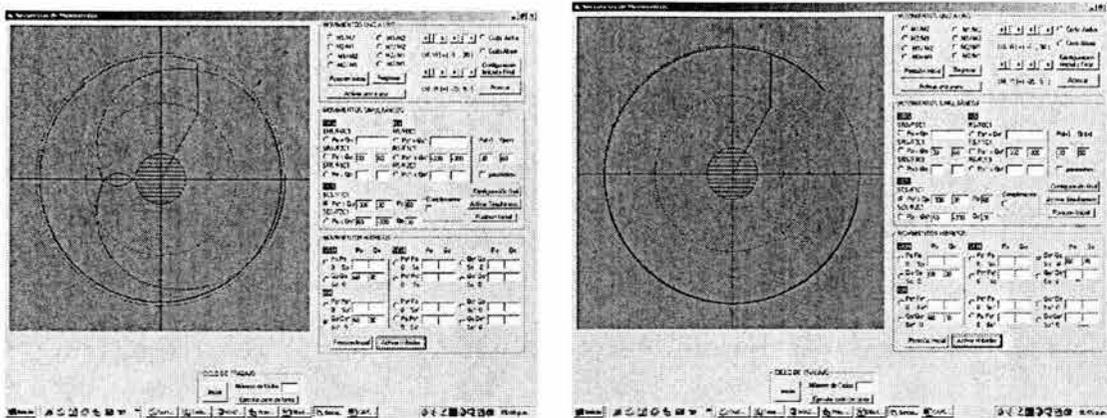
b)

$MCH^{F_1}_{M_{1,2}, T_{1,1}}$		
T \ M	M_1	M_2
T_1	-1	-1
T_2	1	0

c)

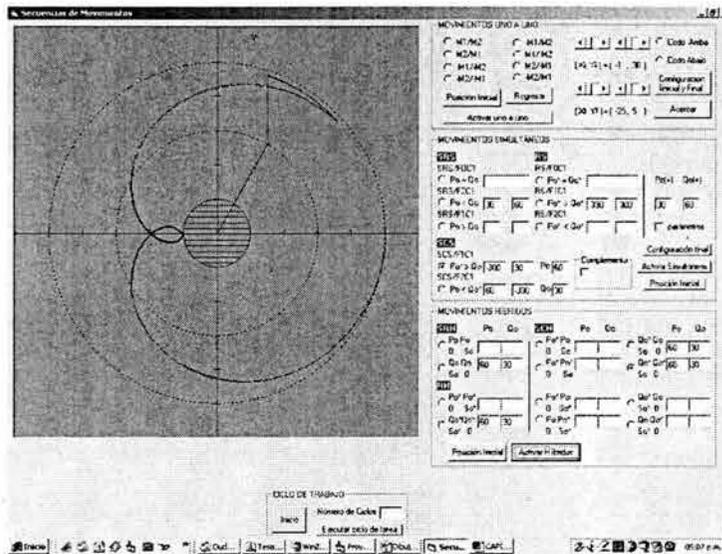
Tabla 3.4 Códigos de los movimientos híbridos: a) complementario, b), c) combinado simple y doble

La figura siguiente muestra las salidas gráficas de los movimientos descritos anteriormente.



a)

b)



c)

Figura 3.5 Salida gráfica de movimientos híbridos de la familia 1: a)complementario, b) combinado simple, c) combinado doble.

Un movimiento híbrido de la familia F_2 es:

$$MCH^{F_2} M_{22, T_{1,2}}$$

La tabla del código es la siguiente:

$MCH^{F2}_{M_{22}, T_{1,2}}$		
T	M	
		M_1
		M_2
	T_1	1
	T_2	0
		-1

Tabla 3.5 Código del movimiento combinado doble de la familia 2

Para finalizar esa sección es conveniente comentar que los movimientos híbridos tienen diferencias importantes:

- 1) El lugar geométrico que recorre el punto terminal del robot es diferente para cada movimiento.
- 2) Los desplazamientos de los eslabones que componen el robot son diferentes para cada movimiento híbrido.
- 3) Existen dos puntos intermedios diferentes, uno para cada familia.

3.6 Algunas consideraciones importantes de los movimientos completos del robot R_0 .

De acuerdo con la caracterización de movimientos realizada en las secciones anteriores, existen [17]:

- 1) 8 movimientos uno a uno.
- 2) 4 movimientos simultáneos.
- 3) 16 movimientos híbridos.

Por tanto, el robot R_0 se puede desplazar de R_0^{ND} a R_0^D de 28 formas posibles, todas ellas diferentes y asociadas con dos configuraciones fijas (R_0^{ND} y R_0^D).

También existen, de acuerdo con [17]:

- a) Dos puntos intermedios para los movimientos uno a uno, cada uno asociado con una familia.
- b) Un número finito de configuraciones intermedias relacionadas con los movimientos simultáneos.
- c) Dos puntos intermedios para los movimientos híbridos, cada uno asociado con una familia. Estos puntos intermedios son diferentes a los relacionados con los movimientos uno a uno.
- d) La configuración de familias para cada tipo de movimientos fue diferente.

Los movimientos completos del robot serán utilizados en los siguientes capítulos de este trabajo de tesis para modelar y simular un problema de evasión de obstáculos en el plano.

CAPÍTULO IV

EVASIÓN MEDIANTE SECUENCIAS

Introducción

Considerando las restricciones e hipótesis fijadas en el capítulo I, en el presente capítulo, se establecerán las características fundamentales de los elementos relacionados con el problema de la evasión de obstáculos; es decir, se realiza una caracterización de las propiedades cinemáticas del robot R_0 , los requerimientos de las tareas, el área de trabajo y, las características del objeto que invade el área de trabajo.

En este capítulo, se aplicarán algunas caracterizaciones de las secuencias de movimientos de un robot de 2 GDL establecidas en el capítulo III [17]. Las secuencias son utilizadas con el objetivo de definir trayectorias de evasión de obstáculos presentes dentro del área de trabajo del robot, también se establecen las configuraciones admitidas por el robot mediante la aplicación de los modelos cinemático directo e inverso desarrollados en el capítulo II.

Para un punto cualquiera de la ruta τ de trabajo, se determinan las configuraciones que puede admitir el robot, y se establecen las posiciones de los obstáculos cuando son tangentes al eslabón L_1 y al eslabón L_2 , dichas posiciones se consideran como el límite en donde el robot y el objeto solo se tocan en un punto. Al generalizar sobre todos los puntos de τ , se establece el lugar geométrico de los centros de todos los objetos que guardan una posición de tangencia respecto a los eslabones L_1 y L_2 del robot, esto, permitirá establecer una serie de criterios para utilizar en la definición de una nueva ruta libre de obstáculos.

Al relacionar las coordenadas del lugar geométrico de los centros de todos los objetos que guardan una posición de tangencia respecto a los eslabones L_1 y L_2

del robot y con las coordenadas que definen la posición del centro de un obstáculo arbitrario, se establecen 5 criterios para lograr la evasión.

Sobre la base de los cuatro primeros criterios, se determinan secuencias de movimientos de los actuadores y la configuración adoptada por el robot en su nueva trayectoria libre de obstáculos.

En el capítulo siguiente se abordará particularmente el análisis del quinto criterio, para definir una nueva trayectoria de evasión, la cual es paralela al contorno del obstáculo.

4.1 Algunas consideraciones fundamentales para la planificación de trayectorias

Para planificar los movimientos de un robot, sobre un área de trabajo libre de obstáculos, es necesario establecer un algoritmo de control que tome en cuenta las siguientes consideraciones:

- i) Propiedades cinemáticas del robot R_o .
- ii) Requerimientos de las tareas.
- iii) Área de trabajo (A_T) del robot.
- iv) Las características del objeto O_b .

Una caracterización mínima de los aspectos anteriores es requerida para poder establecer un esquema de planeación de los movimientos, y así garantizar la capacidad cinemática del mecanismo en presencia de obstáculos y la satisfacción de los requerimientos de las tareas. En otras palabras, la información generada sobre los cuatro puntos anteriores es fundamental para la planeación de las rutas de trabajo alternativas antes de ejecutar los movimientos necesarios para la realización del trabajo.

La ejecución de las rutas alternas establecidas para evitar la colisión, están directamente relacionadas con las propiedades cinemáticas del robot; es decir, se requiere que el robot sea capaz de alcanzar las configuraciones correspondientes, usando para ello secuencias uno a uno, simultáneas o híbridas adecuadas de movimientos en los actuadores. En este capítulo se analiza el caso correspondiente a la evasión de obstáculos a través de secuencias de movimientos complementarios, presentándose los casos específicos cuando es posible cubrir toda la trayectoria original de trabajo, y cuando es imposible cubrirla al 100 %, en las siguientes secciones, se discuten los aspectos mencionados anteriormente.

4.1.1 Propiedades cinemáticas del robot

Las propiedades cinemáticas del robot han sido descritas en el capítulo 2, solo se menciona aquí que, de acuerdo con la tarea por realizar, cada uno de los actuadores puede proporcionar un giro de 360° tanto en sentido positivo como negativo [17,18].

4.1.2 Requerimientos de las tareas

La naturaleza de las tareas por realizar, determinan no solo la ruta por seguir por el órgano efector, sino también la orientación que debe tener en cada punto del recorrido. En el caso de estudio presente, no es considerada la orientación, pues se supone que la tarea solo requiere que el órgano efector se encuentre perpendicular con el plano de trabajo en todo punto de la trayectoria; es decir, la tarea por realizar es la del tipo correspondiente al de la aplicación de pegamento a lo largo de un segmento rectilíneo, particularmente, el órgano efector del robot

tiene la tarea de recorrer la trayectoria τ en un movimiento cíclico de ida y vuelta, según se muestra en la figura 4.1, sin tomar en cuenta el tiempo.

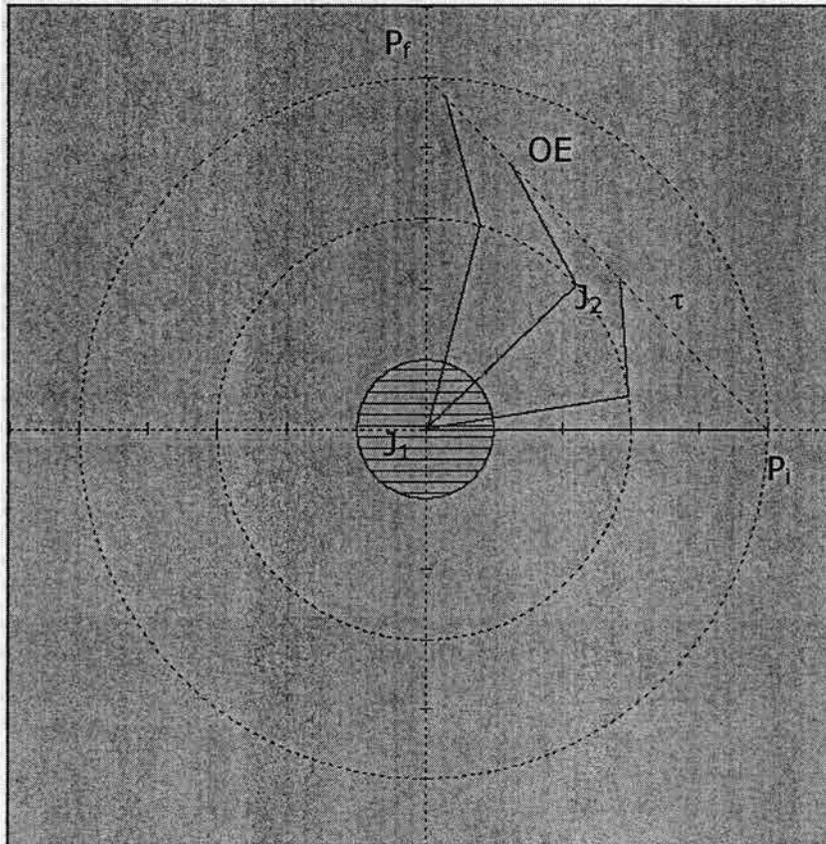


Figura 4.1 Ruta τ de trabajo

Los símbolos mostrados en la figura anterior, se describen de la manera siguiente:

- P_i : es el punto inicial de la trayectoria.
- P_f : es el punto final de la trayectoria.
- OE : es el órgano efector.
- J_1 : articulación en la base.
- J_2 : articulación del codo.
- τ : ruta de trabajo.

De acuerdo con la tarea del robot descrita anteriormente, se tienen las siguientes restricciones de funcionamiento del robot:

- a) No se requiere control sobre la orientación del órgano efector.
- b) Se requiere una ruta rectilínea con dos puntos arbitrarios que indican la configuración inicial (P_i) y final (P_f) del robot.
- c) La tarea debe ejecutarse cíclicamente sin considerar el tiempo.

4.1.3 Área de trabajo

El área de trabajo definida por el robot R_0 se observa en la figura 4.2, la cual corresponde a una circunferencia de radio $R_{R_0} = L_1 + L_2$, esta es posible cubrirla debido al movimiento de los actuadores ubicados en cada una de las uniones (J_1 y J_2) [17], dichos actuadores pueden generar un movimiento angular de 360° en ambos sentidos (levógiro y dextrógiro).

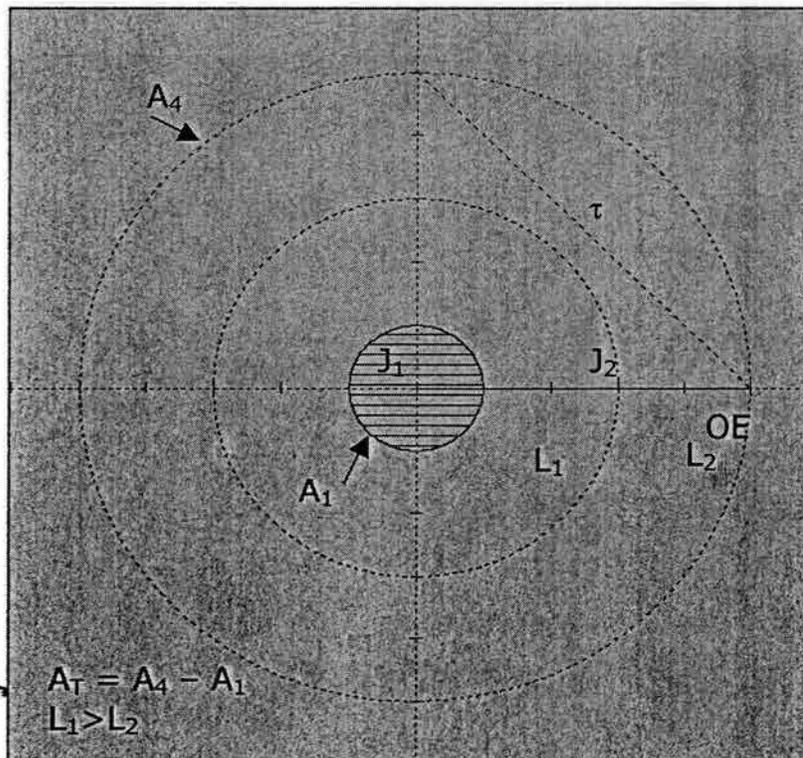


Figura 4.2 Área Total A_T de trabajo con R_0 en posición de inicio

En el caso particular del problema analizado en este capítulo, es conveniente precisar el área de trabajo real que ocupa el robot; es decir, el área donde se localiza el robot de acuerdo con la tarea por realizar, esta área depende de las configuraciones que puede adoptar el multicuerpo para ejecutar la tarea requerida.

De acuerdo con la tarea descrita anteriormente, se requiere cubrir la ruta τ en un movimiento de ida y vuelta por lo que la manera de precisar el área de trabajo real ocupada por el robot, es ubicar la posición adoptada para acceder a cada uno de los puntos de dicha trayectoria, considerando a su vez las diferentes configuraciones correspondientes a cada punto. En la figura 4.3 se observan las dos posibles configuraciones del multicuerpo para tocar un punto arbitrario de la ruta τ .

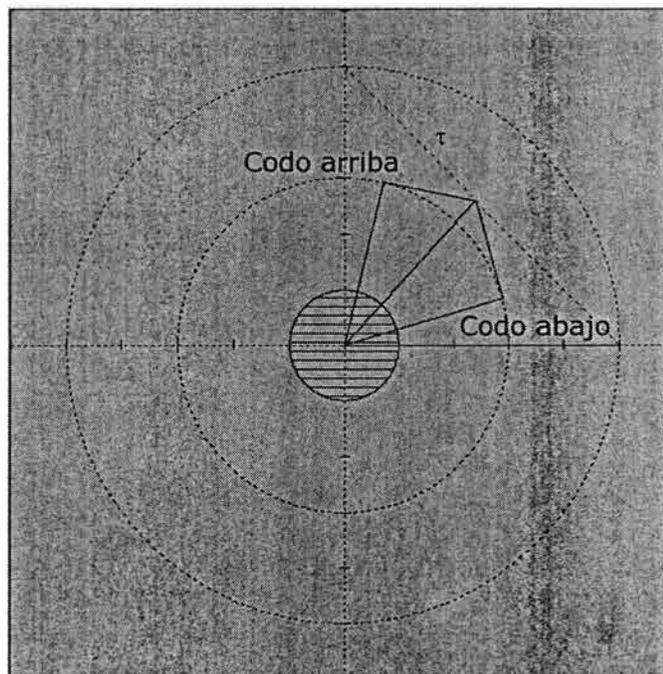


Figura. 4.3 Configuración codo arriba y codo abajo.

Los extremos de la ruta τ son alcanzados cuando el multicuerpo está completamente extendido (alcance máximo $R_{RO} = L_1 + L_2$), cualquier otro punto asociado con la ruta, podrá ser alcanzado por las configuraciones mostradas en la figura anterior.

De acuerdo con lo anterior, es importante señalar que, para establecer el área de trabajo real, bastará con posicionar el multicuerpo en cada uno de los puntos de la trayectoria considerando ambas configuraciones, esto es posible hacerlo experimentalmente si se tiene el robot, o bien a través de un programa de simulación. En el presente caso, el área de trabajo es obtenida a través de la simulación como se muestra en la figura 4.4.

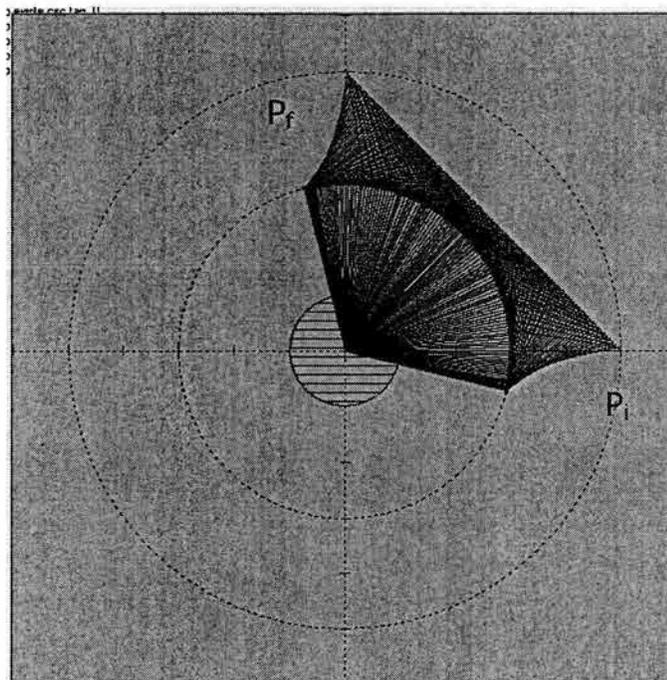


Figura 4.4 Área real (A_R) de trabajo para la trayectoria τ .

De acuerdo con la figura anterior, se observa que cualquier objeto ubicado fuera de esta área no interfiere en las actividades del multicuerpo.

4.1.4 Características del objeto

Las características principales del objeto en estudio, son:

- i) Presenta una geometría de forma circular.
- ii) Se presenta en cualquier posición dentro del área real de trabajo.
- iii) No presenta movimiento.
- iv) Es de un tamaño arbitrario.

Una vez caracterizados los diferentes elementos que determinan el estado del sistema robot-obstáculo, se tiene información suficiente para determinar las posibles situaciones que se pueden presentar en relación con la presencia del objeto y las condiciones de colisión y evasión.

De acuerdo con el área de trabajo real mostrada en la figura 4.4, se pueden hacer las siguientes consideraciones:

1).- El objeto puede quedar fuera del área real de trabajo; es decir, no existe interferencia entre el robot R_o y el objeto, y por lo tanto la ruta τ puede ser recorrida totalmente. Este estado puede ser representado de la siguiente manera [17]:

$$O_b \cap A_R = \emptyset$$

donde: O_b : representa al obstáculo y A_R : representa el área real de trabajo.

2).- La configuración inicial del robot R_o es la posición de referencia (home) por lo que se considera que el objeto no puede localizarse sobre el eje X; es decir:

$$" R_{oi} \text{ indica la posición inicial del multicuerpo y se puede escribir: } R_{oi} \cap O_b = \emptyset "$$

Esto indica que el área bajo el eje X es descartada como zona de presencia del obstáculo, en otras palabras, se considera que una posición límite del obstáculo será cuando esté sobre el eje X sin intersectarlo y dentro del área real de trabajo; es decir:

con base en las anteriores consideraciones, la figura 4.4 ha sido reducida a la mostrada en la figura 4.5.

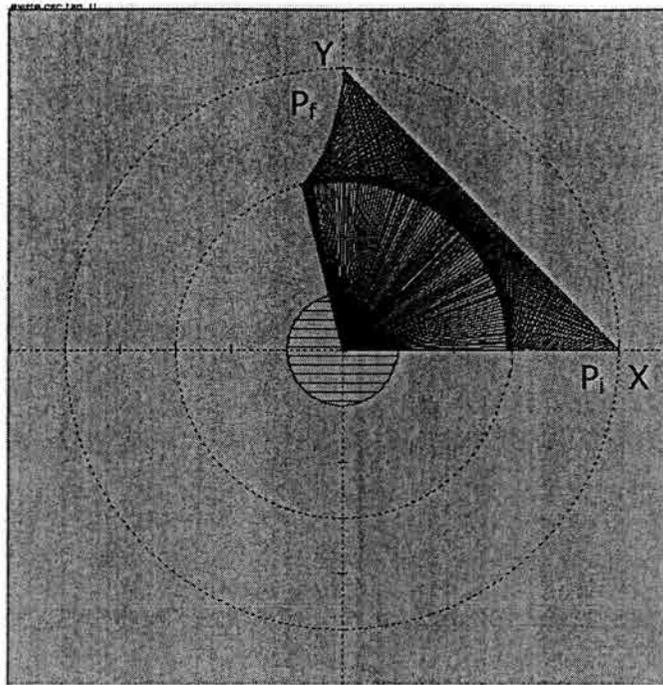


Figura 4.5 Zona reducida de presencia del obstáculo, respecto al eje X

3).- Un análisis similar con respecto a la configuración final de R_o ; es decir, sobre el eje Y, conduce a lo siguiente: para tocar el punto final (sobre los extremos del lugar geométrico) de la trayectoria, R_o debe estar en su máxima extensión; es decir en dicho punto no existen dos formas diferentes de accederlo como en el resto de τ (codo arriba y codo abajo), por lo que un obstáculo que intersecte al eje Y impedirá que dicho punto sea inalcanzable, pues cualquier otra posición de los eslabones L_1 y L_2 implica un alcance menor al caso analizado. De otra manera, si el obstáculo no toca al eje Y, entonces el órgano efector podrá acceder al punto final de τ , por lo que, en este trabajo se considera que la relación: de $O_b \cap R_{of} = \emptyset$ implica que el área a la izquierda del eje Y, no sea considerada y se descarta como zona de presencia del obstáculo, en otras palabras, una segunda posición límite del obstáculo será cuando se encuentre a la derecha del eje Y dentro del área real de trabajo y sin intersectarlo.

En la figura 4.6 se muestra la zona de presencia del objeto con respecto a los ejes X e Y, observándose que su presencia, estará ubicada en el primer cuadrante sin intersectar sus límites; es decir, los ejes X e Y.

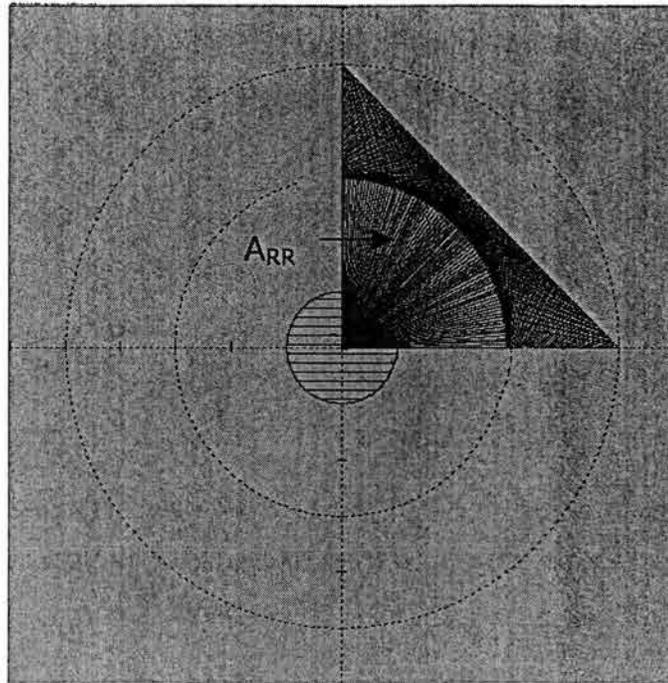


Figura 4.6 Área real reducida (A_{RR}) de presencia del obstáculo, respecto a lo dos ejes

De acuerdo con los tres incisos anteriores, es posible establecer las siguientes relaciones:

- i) $\delta O_b \cap \delta A_{RR} = \emptyset$: objeto totalmente ajeno con el área de trabajo real reducida.
- ii) $O_b \subseteq A_{RR}$: objeto totalmente contenido en el área de trabajo real reducida.
- iii) $O_b \subset A_{RR}$: objeto parcialmente contenido en el área de trabajo real reducida.

En el primer caso el objeto no interfiere con R_o , por tanto la trayectoria puede ser recorrida libremente, y el desplazamiento del multicuerpo puede ser usando tanto la configuración de codo arriba como codo abajo.

En el segundo caso el obstáculo está totalmente contenido en el área real reducida (A_{RR}) por lo que el porcentaje de la trayectoria recorrida dependerá de la ubicación del obstáculo; es decir, depende de la proximidad del obstáculo con la base y al órgano efector del robot, o bien con respecto al origen del sistema de referencia y la trayectoria τ .

En el tercer caso se considera la situación cuando el obstáculo se intersecta con la trayectoria τ , y bajo estas condiciones, debe discriminarse el uso de las configuraciones del codo, con el objetivo de que el órgano efector se desplace sobre la ruta de trabajo, cubriendo la mayor extensión posible.

4.2 Caracterización de las posiciones relativas entre el robot y el obstáculo

Partiendo de las tres condiciones descritas en la sección anterior, se establecerán criterios para determinar los movimientos requeridos por el robot para evadir los posibles obstáculos, manteniendo en lo posible, el órgano efector sobre la trayectoria [17].

La proposición $O_b \cap A_{RR} = \emptyset$ implica que el objeto es totalmente ajeno con el área de trabajo real reducida, por lo que no existe colisión entre el robot y el obstáculo, en este caso, el multicuerpo puede ejecutar su tarea libremente, siendo necesario determinar los criterios de evasión o de no colisión en los dos restantes, es decir cuando se cumplen:

ii) $O_b \subseteq A_{RR}$

iii) $O_b \subset A_{RR}$

La segunda condición será considerada para lograr la evasión, mediante las secuencias de movimientos permisibles por los actuadores, en tanto en la

tercera, se aplicará un análisis para efectuar una evasión a través de una nueva trayectoria paralela al contorno del obstáculo; es decir, por medio de un barrido de frontera.

4.2.1 Determinación de condiciones para las posiciones límites entre R_o y O_b

El algoritmo de control que permita la determinación de una trayectoria libre de obstáculos, será obtenido a través de la determinación previa de las posiciones límites (tangencia entre el mecanismo y el obstáculo) del objeto para cada una de las configuraciones del mecanismo en cada uno de los puntos de la trayectoria τ ; es decir, la estrategia será establecer la posición del centroide del obstáculo que implica una tangencia entre el robot y el obstáculo, lo cual determina la posición límite relativa entre ambos; es decir, donde solo se intersectan en un punto.

La situación anterior se muestra en la figura 4.7 donde se tienen dos circunferencias representativas de los obstáculos Ob_1 y Ob_2 con centros en $C_1 (h_1, k_1)$ y $C_2 (h_2, k_2)$ que definen la posición límite entre el robot y el objeto al acceder a un punto sobre la trayectoria τ . Por otro lado, $C_1 (h_1, k_1)$ es el centro del objeto cercano a la base inercial cartesiana, mientras que $C_2 (h_2, k_2)$ es el centro del objeto cercano al órgano efector, ambos establecen la posición de tangencia con respecto a los eslabones L_1 y L_2 .

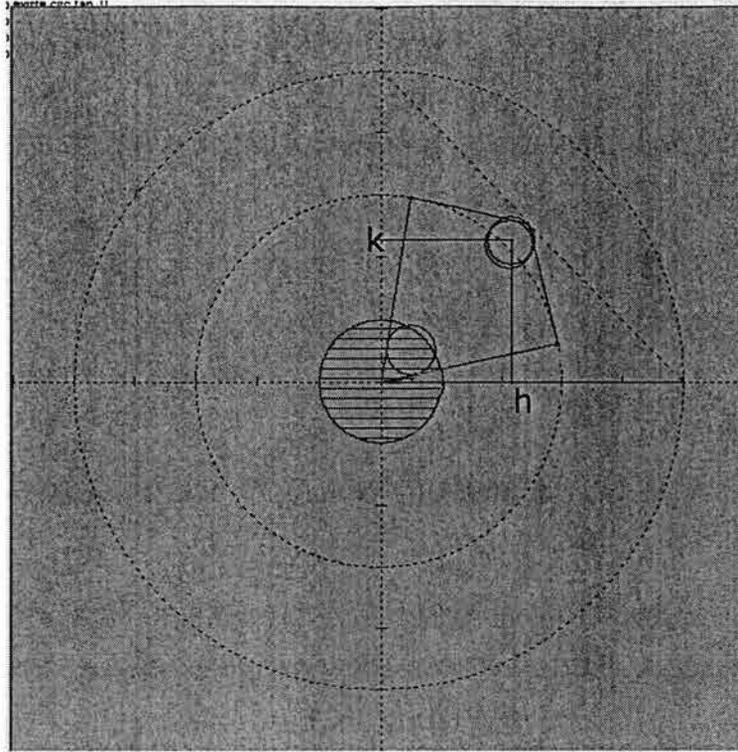


Figura 4.8 El eslabón L_2 es secante al obstáculo Ob

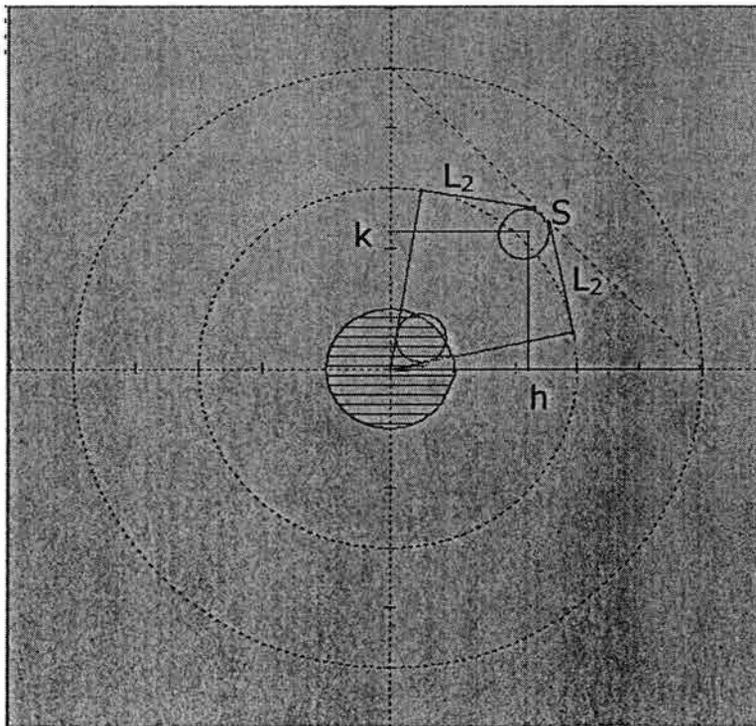


Figura 4.9 Eslabón L_2 tangente al obstáculo

Si el centro $C(h, k)$ del objeto es más cercano al origen $O(0, 0)$, entonces será necesario mover el eslabón L_1 hasta alcanzar la tangencia entre R_o y O_b lo cual igualmente implica la existencia de un segmento sobre τ que no podrá ser recorrido.

Los criterios empleados para la determinación de la posición relativa entre el obstáculo y el robot, se obtienen de las características geométricas mostradas en la figura 4.10.

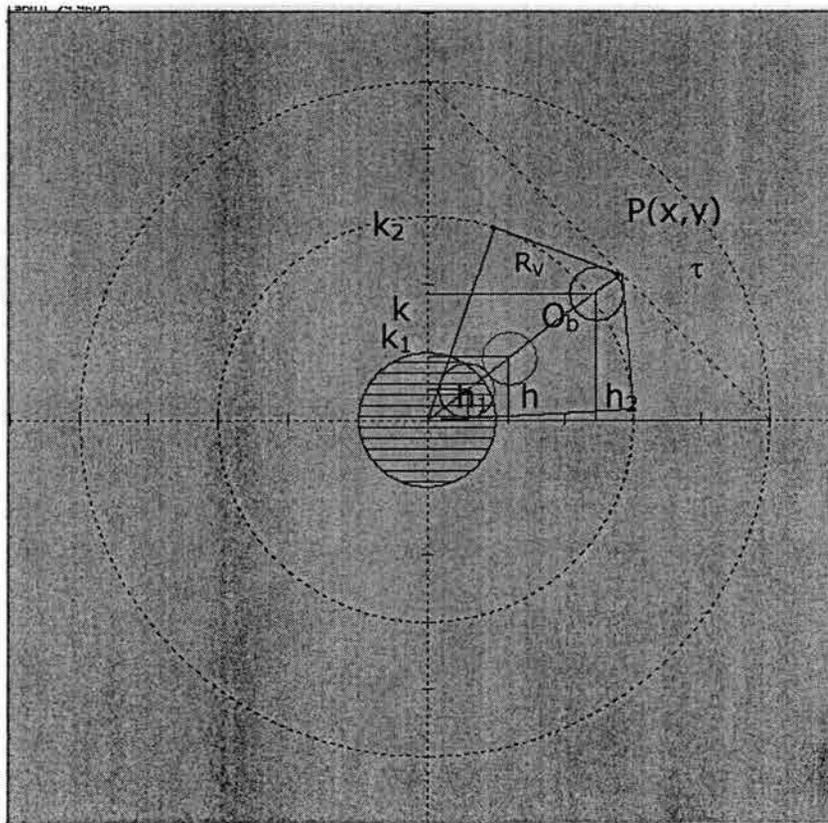


Figura 4.10 Ubicación aleatoria del obstáculo O_b

Es decir, para un punto $P(x, y)$ sobre τ , se determinan las posiciones extremas que puede ocupar el obstáculo, permitiendo aún que el órgano efector alcance la

trayectoria, para posteriormente comparar la posición real del obstáculo y definir el criterio de evasión.

De acuerdo con la figura 4.10, la posición del objeto se determina a través de las coordenadas (h, k) del centro C , y es posible conocer:

- 1) La ecuación del segmento Rv que une el origen del sistema de coordenadas con τ y pasa por $C(h,k)$.
- 2) Las coordenadas (h_1, k_1) y (h_2, k_2) centros de los obstáculos en su posición de tangencia.

1) El segmento Rv , el cual tiene uno de sus extremos en el origen $(0,0)$, y el segundo extremo sobre τ , es un eje de simetría del obstáculo. El extremo sobre τ puede ser accesado a través de las dos configuraciones del multicuerpo.

La ecuación punto-pendiente de una recta es de la forma:

$$y - y_1 = m(x - x_1) \tag{4.1}$$

Aquí, $(x_1, y_1) = (0, 0)$ o $(x_1, y_1) = (h, k)$ y $m = \frac{k}{h}$

Nótese que si $m = \frac{k}{h}$ la expresión (4.1) se escribe como:

$$\left(\frac{x}{h}\right)x - y + \left[y_1 - \left(\frac{x}{h}\right)x_1\right] = 0 \tag{4.2}$$

Obsérvese que si $x_1 = 0$ y $y_1 = 0$ la ecuación (4.2) toma la forma particular:

$$\left(\frac{x}{h}\right)x - y = 0 \tag{4.3}$$

Por otro lado, la ecuación asociada con la trayectoria τ es:

$$y - y_1 = [(y_2 - y_1)/(x_2 - x_1)](x - x_1) \quad (4.4)$$

donde: $m = [(y_2 - y_1)/(x_2 - x_1)]$

Aquí, las coordenadas (x_1, y_1) , (x_2, y_2) representan los extremos de τ .

Si $(x_1, y_1) = (50,0)$ y $(x_2, y_2) = (0,50)$, entonces $m = -1$

Sustituyendo m y (x_1, y_1) en la expresión (4.4) tenemos la ecuación de la ruta τ :

$$x + y = 50 \quad (4.5)$$

Finalmente, resolviendo el sistema formado por (4.3) y (4.5) se determinan las coordenadas del punto $P(x,y)$ sobre τ y extremo del segmento R_v , las cuales son:

$$x = (50h) / (k + h)$$

$$y = (50k) / (k + h)$$

De acuerdo con la figura 4.10, el punto $P(x,y)$ sobre τ puede ser accesado por el robot a través de ambas configuraciones, ya que R_v es un eje de simetría del objeto, por lo que ahora se requiere establecer los centros (h_1, k_1) y (h_2, k_2) de los objetos tangentes a los dos eslabones L_1 y L_2 , respectivamente para ambas configuraciones.

2) Las coordenadas (h_1, k_1) y (h_2, k_2) de los centros de los objetos en su posición límite se obtienen aplicando el concepto de distancia entre un punto y una recta bajo el siguiente procedimiento:

De acuerdo con la figura 4.10, las coordenadas (h_1, k_1) pueden determinarse resolviendo un sistema de dos ecuaciones con dos incógnitas, en este caso las ecuaciones se describen de la manera siguiente:

- a) Ecuación 1: distancia entre el punto (h_1, k_1) y el segmento L_1 .
- b) Ecuación 2: ecuación de Rv que es satisfecha por el punto (h_1, k_1) .

Para establecer la Ecuación 1 se aplica la siguiente expresión:

$$D = \frac{(Ax_1 + By_1 + C)}{\pm \sqrt{A^2 + B^2}} \quad (4.6)$$

Aquí:

$Ax + By + C = 0$ es la ecuación de una recta, en su forma general,

$P(x_1, y_1)$ son las coordenadas de un punto no perteneciente a la recta y

D es la distancia perpendicular entre el punto $P(x_1, y_1)$ y la recta $Ax + By + C = 0$

Para el caso estudiado en este capítulo, la ecuación de la recta que representa al eslabón L_1 es de la forma:

$$m_1x - y = 0 \quad (4.7)$$

con $m_1 = y_c / x_c$

y x_c es la abscisa del codo y y_c es la ordenada del codo.

Las coordenadas del punto P son (h_1, k_1) y la distancia $D = r$ indica la longitud del radio del objeto.

Por tanto, al sustituir D por r , (x_1, y_1) por (h_1, k_1) y la expresión (4.7) en la expresión (4.6) se obtiene la ecuación 1 buscada:

$$r = \frac{(m_1 h_1 - k_1)}{\pm \sqrt{m_1^2 + 1}}$$

o, equivalentemente:

$$m_1 h_1 - k_1 = \pm r \sqrt{m_1^2 + 1} \quad (4.8)$$

Por otro lado, la ecuación 2 de interés, representa al segmento Rv que contiene al punto (h_1, k_1) , y al origen $(0,0)$, por lo que es de la forma :

$$y = m_2 x \quad (4.9)$$

donde $m_2 = k/h$

Sustituyendo el punto (h_1, k_1) en la ecuación (4.9), se tiene:

$$k_1 = m_2 h_1 \quad (4.10)$$

Resolviendo el sistema formado por las expresiones (4.8) y (4.10) se obtienen las coordenadas del centro del objeto tangente al eslabón L_1 :

$$h_1 = \left(\pm r \sqrt{m_1^2 + 1} \right) / (m_1 - m_2) \quad (4.11)$$

$$k_1 = m_2 h_1$$

Es decir, $C_1 (h_1, k_1)$ es el centro de la circunferencia correspondiente con el obstáculo que es tangente al eslabón L_1 tanto en la configuración de codo arriba como de codo abajo, como se muestra en la figura 4.11

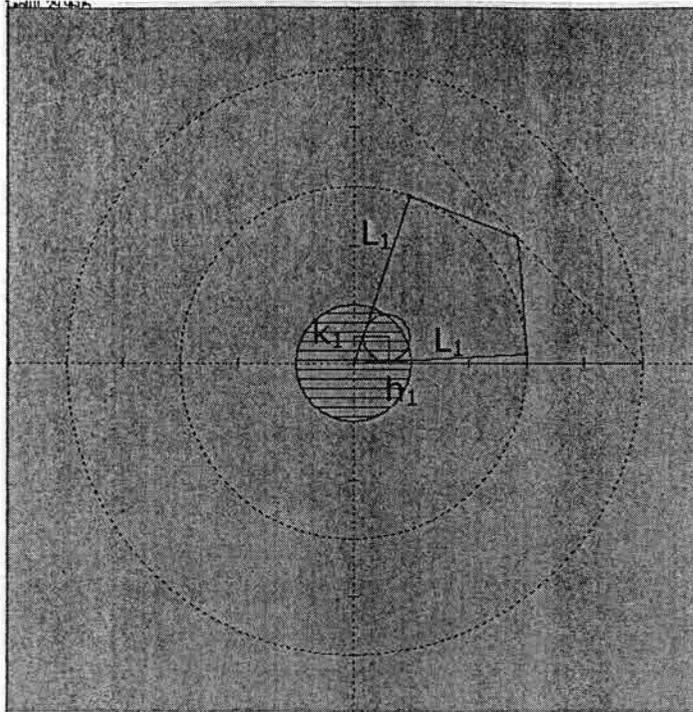


Figura 4.11 obstáculo tangente al eslabón L_1

Para la obtención de las coordenadas de $C_2 (h_2, k_2)$, que representan el centro del obstáculo tangente al eslabón L_2 para las dos configuraciones, se utiliza un procedimiento similar al anterior; es decir, se establecen dos ecuaciones que contengan como incógnitas las coordenadas (h_2, k_2) ; estas ecuaciones son:

$$1) y = m_2x \quad (4.9)$$

$$2) m_4 h_2 - k_2 = \pm r \sqrt{m_4^2 + 1} - (y_1 - m_4 x_1) \quad (4.12)$$

La ecuación (4.12) representa la distancia entre el punto $C_2 (h_2, k_2)$ y la recta que contiene al eslabón L_2 y se ha obtenido aplicando la expresión (4.6).

La expresión (4.6) representa la distancia entre un punto y una recta dados, en este caso el punto es $C_2 (h_2, k_2)$ y la recta es la que contiene al eslabón L_2 .

La recta que contiene al eslabón L_2 es de la forma descrita por la ecuación (4.1):

$$y - y_1 = m_4(x - x_1)$$

o, equivalentemente:

$$m_4x - y + (y_1 - m_4x_1) = 0 \quad (4.13)$$

donde :

$$m_4 = (y_2 - y_1)/(x_2 - x_1)$$

con: $x_1 = (50h)/(k+h)$ y $y_1 = (50k)/(k+h)$

Aquí, (x_1, y_1) son las coordenadas del extremo de Rv obtenidas anteriormente al resolver el sistema con las ecuaciones (4.3) y (4.5), y (x_2, y_2) son las coordenadas correspondientes con el codo del robot. Estas últimas coordenadas, son obtenidas aplicando el modelo de la cinemática inversa; es decir, conociendo la posición del órgano efector (extremo de Rv) se determinan las dos configuraciones correspondientes al codo arriba y codo abajo; además, se obtiene la información del ángulo asociado con cada eslabón (L_1 y L_2), así como la posición de la articulación J_2 .

Utilizando la ecuación (4.13) y aplicando la ecuación (4.6), es posible expresar la distancia entre el punto $C_2 (h_2, k_2)$ y la recta que contiene a L_2 de la siguiente manera:

$$[\{m_4h_2 - k_2 + (y_1 - m_4x_1)\} / \pm \sqrt{m_4^2 + 1}] = r$$

o bien: ,

$$m_4h_2 - k_2 = \pm r \sqrt{m_4^2 + 1} - (y_1 - m_4x_1) \quad (4.12)$$

Resolviendo el sistema formado por las dos ecuaciones (4.9) y (4.12), se obtienen las coordenadas (h_2, k_2) esto es :

$$h_2 = \{\pm r\sqrt{m_4^2 + 1} - (y_1 - m_4 x_1)\} / (m_4 - m_1) \quad (4.14)$$

$$k_2 = m_3 h_2$$

Donde $C_2(h_2, k_2)$ representa el centro del objeto tangente al eslabón L_2 para ambas configuraciones, según se muestra en la figura 4.12.

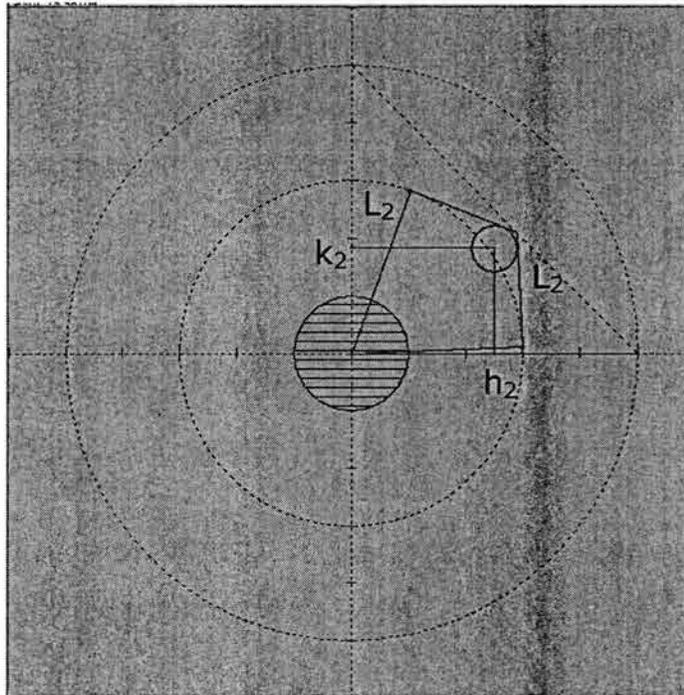


Figura 4.12 Objeto tangente al eslabón L_2

Con las ecuaciones (4.11) y (4.14) se tiene la información de la configuración del robot con la posición límite del objeto; es decir, la posición del objeto que es tangente a los eslabones L_1 y L_2 en las configuraciones de codo arriba y codo abajo. Estas configuraciones serán tomadas en cuenta para distinguir si un objeto en cualquier posición arbitraria dentro del área real restringida de trabajo podrá

evadirse, y además cubrir la trayectoria especificada. La información obtenida de las expresiones (4.11) y (4.14) es: $C_1 (h_1, k_1)$ y $C_2 (h_2, k_2)$ que representan las coordenadas de los centros de los objetos tangentes a los eslabones L_1 y L_2 los cuales se observan en la figura 4.13.

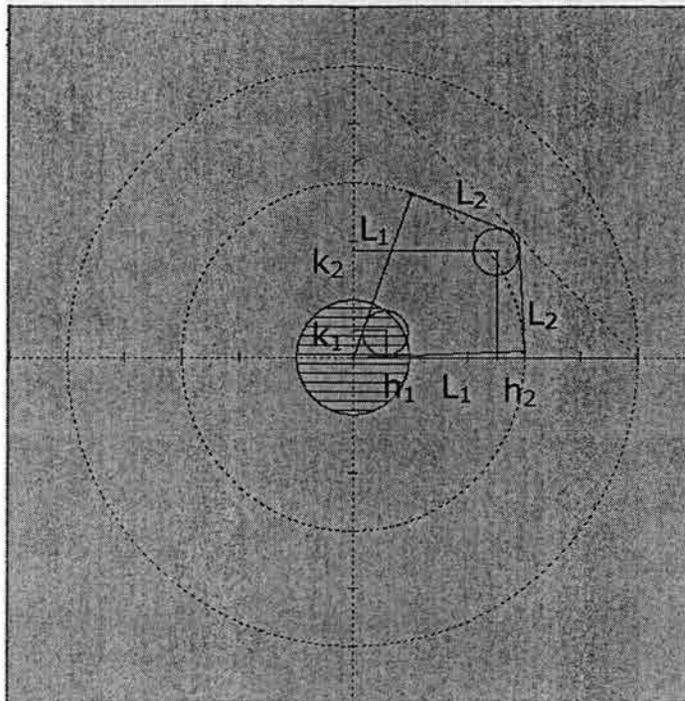


Figura 4.13 Posiciones límites entre el objeto y el robot

Con la anterior sistematización se han establecido las ecuaciones que permiten determinar las posiciones límites entre R_o y O_b dentro de A_{RR} , con solo conocer la posición del objeto. El conocimiento de las coordenadas de $C_1 (h_1, k_1)$ y $C_2 (h_2, k_2)$ permitirá decidir la forma y condiciones de evasión para un caso particular. Si el objeto se presenta en otra región, será necesario calcular las nuevas posiciones límites correspondientes con las tangencias entre R_o y O_b , y esto se requiere para cualquier otra posición del objeto dentro del área de trabajo, por lo que es conveniente fijar el lugar geométrico del centro del objeto para la posición de tangencia entre R_o y O_b , para cada uno de los puntos que forman la ruta de trabajo τ , considerando las dos configuraciones del codo.

La posición de tangencia entre R_o y O_b , representa el límite entre las posiciones de ambos objetos entre sí, sin que exista penetración, en la figura 4.14 se muestra esta situación.

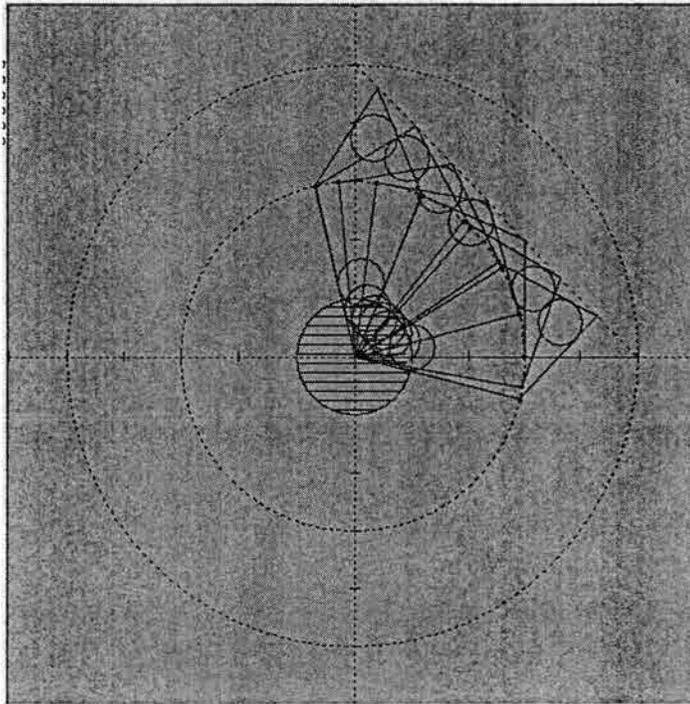


Figura 4.14 Posiciones límites de tangencia

Aplicando las ecuaciones (4.11) y (4.14) para $C_1 (h_1, k_1)$ y $C_2 (h_2, k_2)$, se obtiene el lugar geométrico que indica la posición del centro de los objetos que ocupan la posición límite de tangencia con R_o ; es decir, objetos tangentes a los eslabones L_1 y L_2 , esto se muestra en la figura 4.15. El interior del lugar geométrico (el cual es cerrado) define el área A_p .

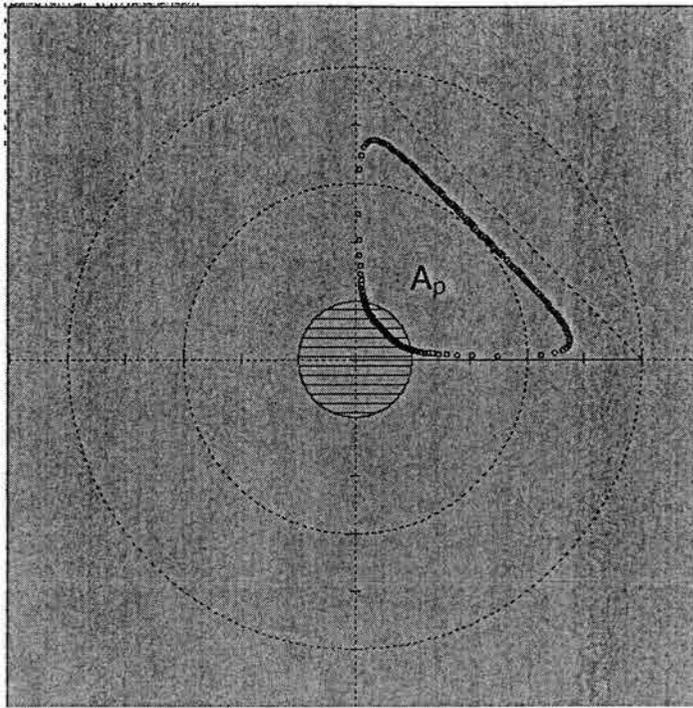


Figura 4.15 Lugar geométrico de C_1 y C_2

En la figura 4.15 se observa que, cualquier objeto cuyo centro quede dentro del área A_p , podrá ser evadido y la trayectoria τ será cubierta totalmente. Por otro lado, cualquier objeto cuyo centro quede fuera de A_p pero dentro de A_{RR} , impedirá que el órgano efector pueda recorrer la ruta τ en forma completa, esto se observa en la figura 4.16.

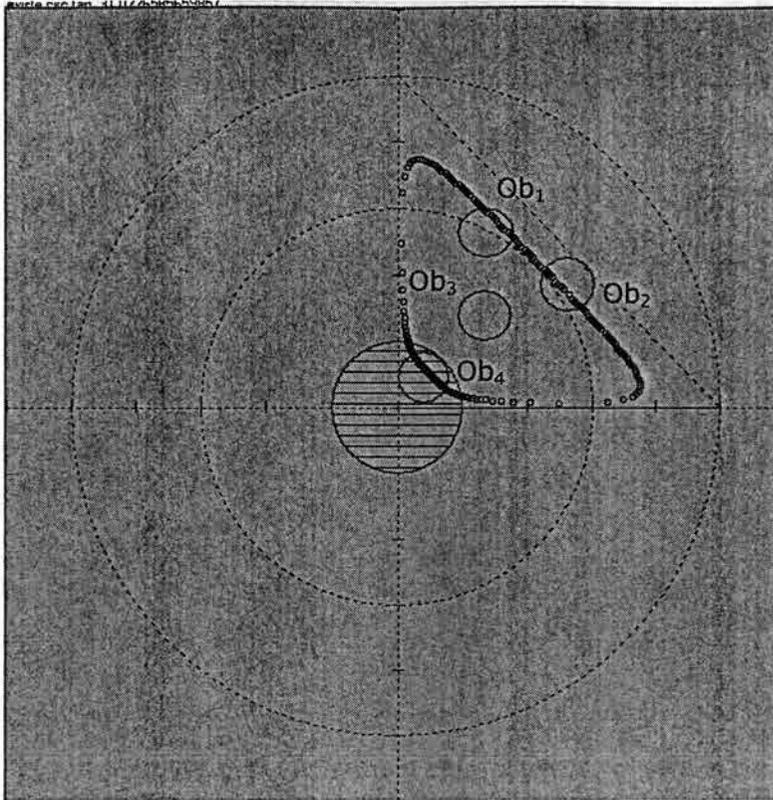


Figura 4.16 Objetos con el centro dentro y fuera de A_p

Los objetos Ob_1 y Ob_3 mostrados en la figura 4.16, pueden ser evadidos cubriendo toda la trayectoria, mientras que los objetos Ob_2 y Ob_4 pueden ser evadidos pero la trayectoria τ no será tocada al 100 %.

4.2.2 Criterios de evasión

En la sección 4.1.4 se establecieron las relaciones:

- i) $\delta O_b \cap \delta A_{RR} = \emptyset$; objeto totalmente ajeno con el área de trabajo real reducida.
- ii) $O_b \subseteq A_{RR}$; objeto totalmente contenido en el área de trabajo real reducida.
- iii) $O_b \subset A_{RR}$; objeto parcialmente contenido en el área de trabajo real reducida.

En esta sección, se especificarán los criterios correspondientes para cada uno de las tres condiciones anteriores con el fin de definir la forma de evasión del obstáculo, para esto se utiliza básicamente, la información del lugar geométrico de los centros de los objetos en su posición tangente a R_0 .

Criterio 1.- Si $h_1 < h < h_2$, entonces la trayectoria τ puede cubrirse totalmente utilizando ángulos complementarios y las configuraciones de codo arriba y codo abajo mencionadas en el capítulo II.

Aquí:

(h, k) : son las coordenadas del centro del objeto, en posición arbitraria.

(h_1, k_1) : son las coordenadas del centro del objeto en posición tangente al eslabón L1 en ambas configuraciones de codo.

(h_2, k_2) : son las coordenadas del centro del objeto en posición tangente al eslabón L2 en ambas configuraciones de codo.

Criterio 2.- Si $h < h_1$, entonces el eslabón L_1 debe ser movido hasta la posición de tangencia con Ob, por lo que quedará un segmento S sobre τ imposible de tocar. Para alcanzar la parte τ factible, se deberán usar movimientos complementarios, requiriendo además, ambas configuraciones.

Aquí h y h_1 son las mismas del criterio 1.

Criterio 3.- Si $h > h_2$ y $L_{R+r} < R_v$, entonces L_2 debe ser movido hasta la posición de tangencia por lo que quedará un segmento de τ imposible de seguir, y la parte factible de alcanzar será a través de movimientos complementarios y aplicando las dos configuraciones del codo.

Aquí:

h y h_2 son las mismas del criterio 1.

L_R : es la longitud del segmento (OC) definido del origen del sistema cartesiano al centro del objeto.

r : es la longitud del radio del obstáculo.

En este caso la posición del objeto es tal que:

- a) Intersecta el lugar geométrico de J_2 .
- b) No intersecta a τ .

Criterio 4.- Si $h > h_2$ y $L_R - r > L_1$ y $L_R + r \leq R_v + r$, entonces la evasión ya no se hará a través de movimientos complementarios sino a través de una trayectoria paralela a la frontera del objeto. Por tanto, la trayectoria τ no podrá ser tocada en su totalidad y se deberá usar ambas configuraciones de codo.

Aquí :

h y h_2 son las mismas del criterio 1.

L_R y r son los mismos que en el criterio 3.

R_v : segmento definido del origen del sistema de coordenadas, hasta τ y pasa por $C(h,k)$.

En este caso la posición del objeto es tal que:

- a) No intersecta el lugar geométrico de J_2 .
- b) intersecta a τ o no lo intersecta.

Criterio 5. Si $L_R - r > R_v$, entonces la posición del objeto no toca a τ y queda completamente fuera de A_{RR} por lo que el robot podrá ejecutar su operación normalmente.

Aquí :

L_R , r y R_v son los mismos que en el criterio 3.

Como se estableció en las consideraciones de la sección 4.1.4, las posiciones de objetos tangentes a los ejes X e Y no serán tomados en cuenta. La posición del objeto en los cuadrantes II, III, y IV queda fuera del área real de trabajo del manipulador.

La relación $O_b \cap A_{RR} = \emptyset$ que considera al objeto totalmente disyunto al área de trabajo real reducida, es satisfecha por el criterio 5.

La relación $O_b \subseteq A_{RR}$ que considera al objeto totalmente contenido en el área de trabajo real reducida, es satisfecha por los criterios 1, 2 y 3 .

La relación $O_b \subset A_{RR}$ que considera al objeto parcialmente contenido en el área de trabajo real reducida, es satisfecha por el criterio 4 .

4.2.2.1 Parámetros y movimientos asociados con los criterios de evasión

Criterio 1.- Si $h_1 < h < h_2$, entonces la trayectoria τ puede cubrirse totalmente utilizando ángulos complementarios y las configuraciones de codo arriba y codo abajo.

En este criterio, la recta que contiene al radio vector (segmento que une al origen del sistema de coordenadas con el centro del objeto), toca a τ en un punto $P(x, y)$ el cual puede ser alcanzado por el robot en ambas configuraciones del codo, sin tocar al objeto. Esto se observa en la figura 4.17.

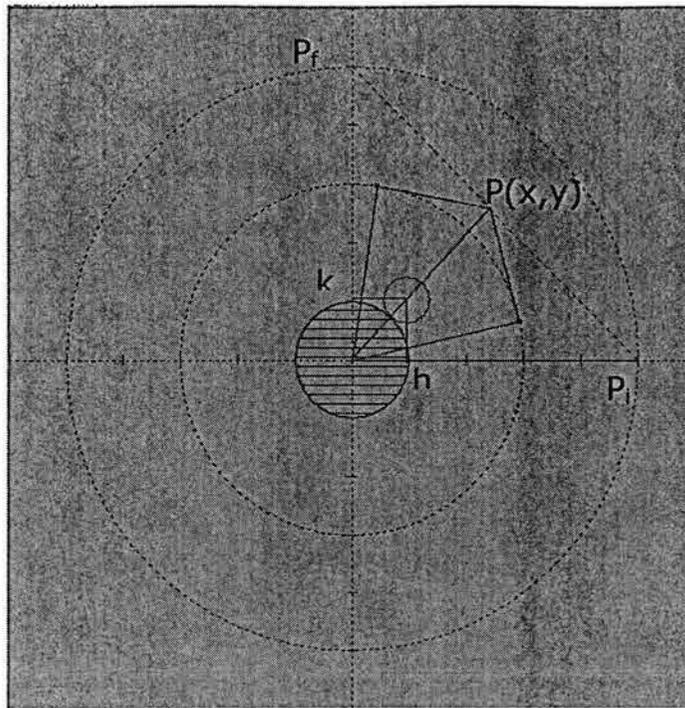


Figura 4.17 Punto P perteneciente a τ accesible en codo arriba y codo abajo

En este caso la trayectoria τ puede ser tocada totalmente por el órgano efector mediante los siguientes movimientos:

Movimientos asociados:

- i) De P_i a P en configuración de codo abajo con secuencias simultáneas.
- ii) De P a P_i en configuración de codo abajo con secuencias simultáneas.
- iii) De P_i a P en configuración de codo arriba con secuencias uno a uno y movimientos complementarios.
- iv) De P a P_f en configuración de codo arriba con secuencias simultáneas.
- v) De P_f a P en configuración de codo arriba con secuencias simultáneas.
- vi) De P a P_i en configuración de codo arriba con secuencias uno a uno.

Cabe señalar que, con el uso de dichos movimientos, será posible que la trayectoria τ se cubra al 100 %.

Criterio 2.- Si $h < h_1$, entonces el eslabón L_1 debe ser movido hasta la posición de tangencia con O_b , por lo que quedará un segmento S sobre τ imposible de tocar. Para alcanzar la parte τ factible, se usarán movimientos complementarios, requiriendo además, ambas configuraciones.

Para determinar las configuraciones de codo arriba y codo abajo, primeramente se determina la posición de tangencia entre el eslabón L_1 y el objeto O_b , y una vez fijada la posición de L_1 , encontrar la posición de L_2 tal que el órgano efector toque a τ . La configuración resultante es mostrada en la figura 4.18.

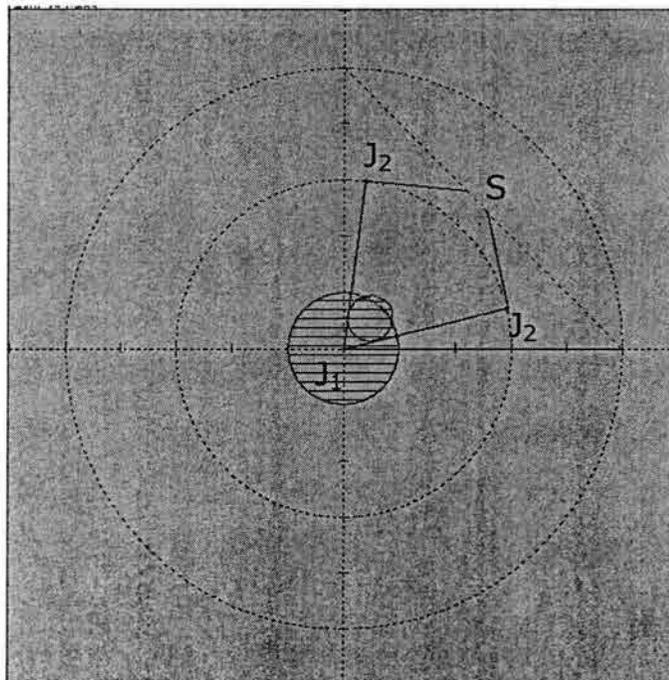


Figura 4.18 Segmento S no accesible al órgano efector

En la figura anterior se observa la existencia del segmento S perteneciente a τ que no podrá ser tocado por el órgano efector, además S es el menor segmento no cubierto en dicha circunstancia, porque cualquier desplazamiento del eslabón

L_1 alejándose del objeto implicará un movimiento del eslabón L_2 incrementando a S debido el acoplamiento entre los eslabones de la cadena cinemática. Para determinar la posición del eslabón L_1 basta con conocer su pendiente y la posición de la junta J_2 , en este caso la pendiente se determina aplicando de nuevo la ecuación de la distancia entre un punto y una recta, de acuerdo a ello, la ecuación del eslabón L_1 es de la forma:

$$y - y_1 = m(x - x_1) \quad (4.15)$$

Dado que el eslabón L_1 parte del origen del sistema de coordenadas, se tiene que $x_1 = y_1 = 0$, por lo que la ecuación (4.15) se puede expresar de la manera siguiente:

$$mx - y = 0 \quad (4.16)$$

Dicha expresión representa la ecuación de la recta que define al eslabón L_1 .

Puesto que se conoce el centro del objeto $C(h, k)$, se aplica la ecuación de la distancia (4.6) entre este punto y la ecuación de L_1 obteniéndose:

$$\left\{ \frac{(mh - k)}{\pm \sqrt{m^2 + 1}} \right\} = r \quad (4.17)$$

Aquí, r es el radio del objeto y (h, k) es el centro del objeto.

Al eliminar el radical y pasando los términos al primer miembro de la ecuación (4.17), se obtiene un ecuación cuadrática en "m" esto es:

$$(h^2 - r^2)m^2 - 2hkm + k^2 - r^2 = 0$$

De donde se obtiene m_1 y m_2 a través de:

$$m = \left(-b \pm \sqrt{b^2 - 4ac} \right) / 2a \quad (4.18)$$

siendo:

$$a = h^2 - r^2, \quad b = -2hkm \quad \text{y} \quad c = k^2 - r^2$$

Al conocer "m", se obtiene la ecuación de la recta que contiene al eslabón L_1 en ambas configuraciones de codo; es decir:

Si se sustituye m_1 y m_2 en la ecuación (4.16) de la recta que contiene a L_1 se tiene que :

$$y = m_1x \quad (4.19)$$

$$y = m_2x \quad (4.20)$$

La expresión (4.19) es la ecuación de la recta que define al eslabón L_2 en codo abajo.

La expresión (4.20) es la ecuación de la recta que define al eslabón L_2 en codo arriba.

La posición del codo (J_2), de acuerdo con la figura 4.19, se obtiene por:

$$x_U = r \cos(\theta_U), \quad y_U = r \sin(\theta_U) \quad (4.21)$$

$$x_D = r \cos(\theta_D), \quad y_D = r \sin(\theta_D) \quad (4.22)$$

Aquí, (x_U, y_U) son las coordenadas del codo en configuración codo arriba.

(x_D, y_D) son las coordenadas del codo en configuración codo abajo.

$$\theta_D = \tan^{-1}(m_2)$$

$$\theta_U = \tan^{-1}(m_1)$$

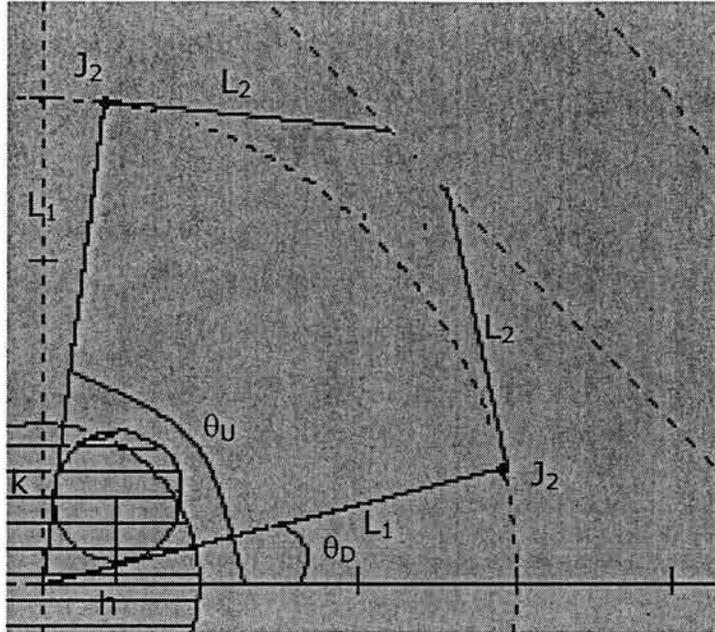


Figura 4.19 Posición tangente entre el eslabón L_1 y el objeto

A continuación se procederá a determinar la posición del eslabón L_2 , manteniendo fija la posición del eslabón L_1 . La nueva posición del eslabón L_2 , debe permitir que el órgano efector quede sobre la trayectoria τ .

En la figura 4.20 se observa que, el eslabón L_2 puede girar sobre J_2 y, por tanto, tocar a τ en dos puntos en cada una de sus configuraciones (P_3 y P_4 en codo abajo y P_1 y P_2 en codo arriba). Los puntos de interés son A y B localizados sobre la trayectoria, los cuales están asociados con P_2 y P_3 . En este caso el segmento AB no será accesible al órgano efector.

3

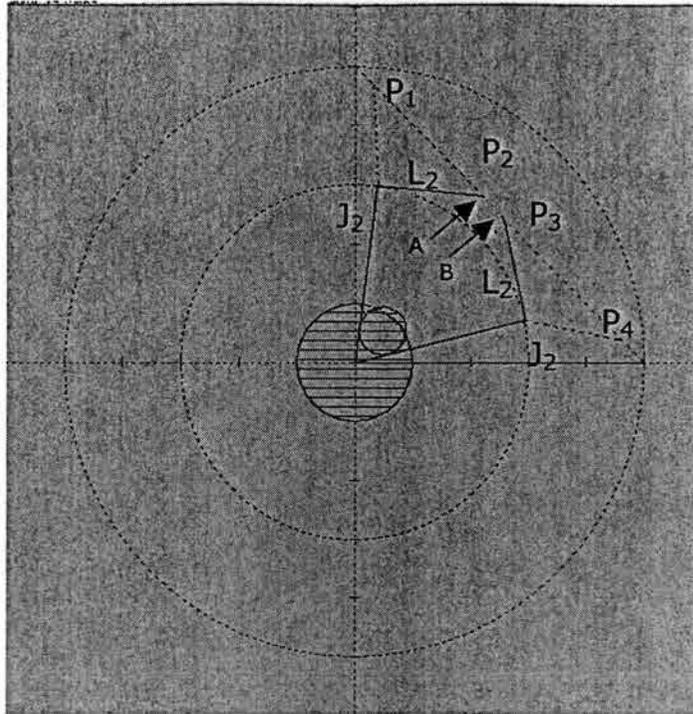


Figura 4.20 Determinación de la posición del eslabón L_2

Como se muestra en la figura anterior, los puntos P_1 , P_2 , P_3 y P_4 representan las intersecciones de el segmento rectilíneo que determina la trayectoria τ y la circunferencia que puede recorrer el órgano efector, cuando el eslabón gira sobre J_2 , por lo que sus coordenadas pueden ser obtenidas resolviendo el sistema de ecuaciones algebraicas:

$$(x - h)^2 + (y - k)^2 = r^2 \tag{4.23}$$

$$x + y = 50 \tag{4.5}$$

La ecuación (4.5) representa la ruta de trabajo del robot, y la expresión (4.23) representa la circunferencia con centro en la junta J_2 y de radio de 20 cm correspondiente a la longitud del eslabón L_2 .

De la expresión (4.5) se tiene que:

$$x = 50 - y$$

y, sustituyendo en la ecuación (4.21) se obtiene la siguiente ecuación de segundo grado en y:

$$2y^2 + (2h - 100 - 2k)y + (50^2 - L_2^2 - 100h + h^2 + k^2) = 0$$

Si:

$$a = 2$$

$$b = 2h - 100 - 2k$$

$$c = 50^2 - L_2^2 - 100h + h^2 + k^2$$

entonces:

$$y = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (4.24)$$

Por tanto:

$$a) \quad x_1 = 50 - y_1 \quad (4.25)$$

$$b) \quad x_2 = 50 - y_2$$

Los modelos para "x" e "y" obtenidos anteriormente, son válidos tanto para codo arriba como para codo abajo. Lo que cambia en cada caso son los valores del punto (h, k) que representa, en este caso, las coordenadas de J_2 y los cuales se observan en la figura 4.19, y se determinan mediante (4.21) y (4.22). Aquí se han obtenido los puntos sobre τ que definen la posición de tangencia entre R_o y O_b .

En este caso la trayectoria τ no puede ser tocada totalmente por el órgano efector, y la nueva ruta de trabajo se recorre mediante los siguientes movimientos:

Movimientos asociados:

- 1) De P_1 a P_3 en configuración de codo abajo con secuencias simultáneas.
- 2) De P_3 a P_1 en configuración de codo abajo con secuencias simultáneas.
- 3) De P_1 a P_2 en configuración de codo arriba con secuencias uno a uno y movimientos complementarios.
- 4) De P_2 a P_f en configuración de codo arriba con secuencias simultáneas.
- 5) De P_f a P_2 en configuración de codo arriba con secuencias simultáneas.
- 6) De P_2 a P_1 en configuración de codo arriba con secuencias uno a uno.

Cabe señalar que, con el uso de dichos movimientos, será posible evadir al obstáculo y que la ruta original τ no podrá ser recorrida totalmente por el órgano efector. La longitud del segmento de τ no tocado por el órgano efector será determinado por la distancia entre los puntos P_2 y P_3 mostrados en la figura 4.20.

Criterio 3.- Si $h > h_2$ y $L_R+r < R_v$, entonces L_2 debe ser movido hasta la posición de tangencia por lo que quedará un segmento de τ imposible de seguir, y la parte factible de alcanzar será a través de movimientos complementarios y aplicando las dos configuraciones del codo.

En este criterio se realiza un análisis similar al realizado para el criterio anterior. En este caso por la posición del objeto, el eslabón L_2 debe ser movido hasta alcanzar la tangencia con el objeto, lo cual igualmente, implicará que un segmento de la trayectoria τ no sea accesible al órgano efector. En la figura 4.21 se muestra esta situación.

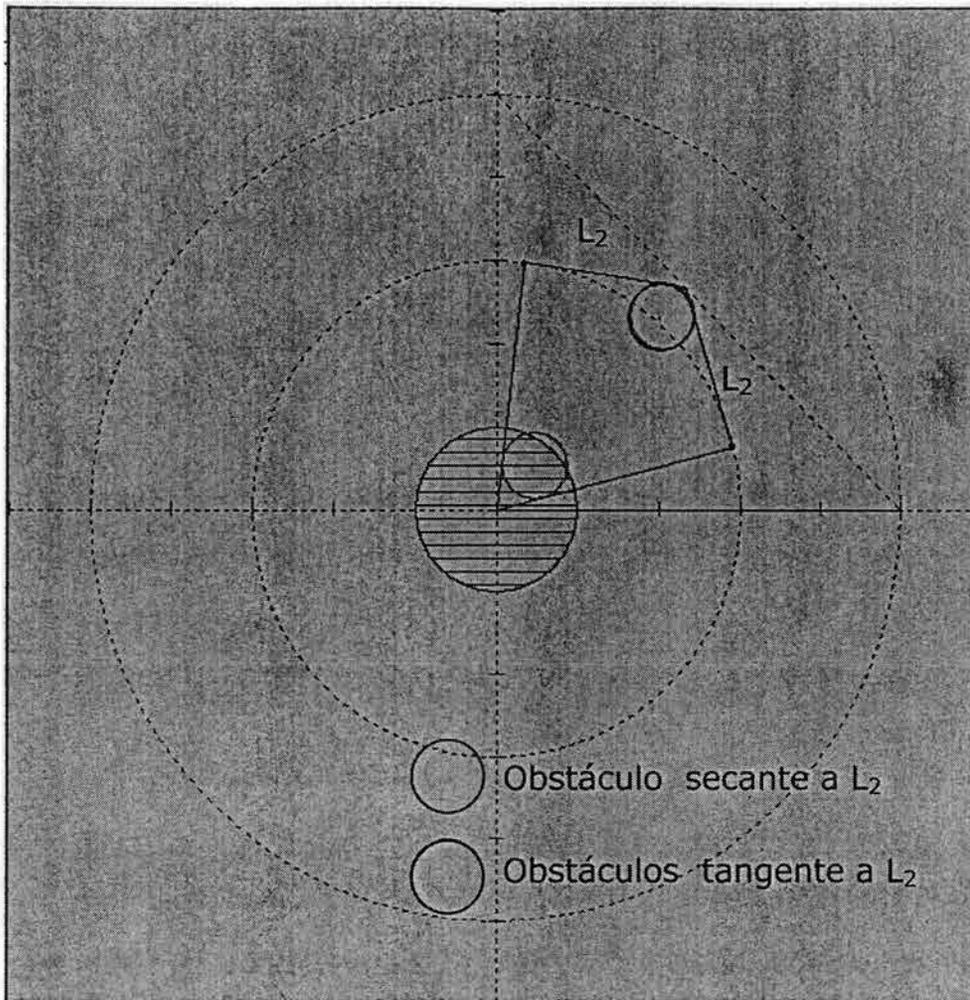


Figura 4.21 Eslabón L_2 en posición secante con el obstáculo

De acuerdo con la figura 4.22, existen dos puntos P_1 y P_2 sobre τ accesibles al órgano efector donde el eslabón L_2 sea tangente con el objeto. Si los puntos P_1 y P_2 son determinados, entonces se tendrán las configuraciones de codo del robot aplicando la cinemática inversa.

Para calcular las coordenadas de los puntos $P_1(x_1, y_1)$ y $P_2(x_2, y_2)$ se aplica el siguiente procedimiento:

1) Determinar las coordenadas de los puntos P_1 y P_2 que pertenecen a τ , tal que la distancia del centro $C(h, k)$ del objeto al eslabón L_2 sea igual al radio del obstáculo.

Si nuevamente se aplica la fórmula de la distancia (4.6) entre un punto y una recta, donde para este caso, la ecuación representa a la recta que contiene al eslabón L_2 , y el punto es el centro del objeto, entonces se tiene que la ecuación de la recta que contiene al eslabón L_2 , tal que, el órgano efector está sobre el punto $P(x, y)$ (lo cual se muestra en la figura 4.22), es de la forma de la expresión (4.1), la cual, puede ser representada de forma equivalente de la siguiente manera:

$$mx - y + y_1 - mx_1 = 0 \quad (4.26)$$

La pendiente "m" del eslabón L_2 , de acuerdo con la figura 4.21, está dada por:

$$m = \frac{(y_{OE} - y_{CD})}{(x_{OE} - x_{CD})} \quad (4.27)$$

Además :

$$(x_1, y_1) = (x_{OE}, y_{OE}) \text{ o } (x_1, y_1) = (x_{CD}, y_{CD})$$

Aquí, (x_{OE}, y_{OE}) son las coordenadas del órgano efector sobre τ , y (x_{CD}, y_{CD}) son las coordenadas del codo según la expresión (4.22).

Si el punto considerado es $C(h, k)$ y la ecuación de la recta es la expresión (4.26), entonces al aplicar la ecuación (4.6) de la distancia, se obtiene :

$$\left\{ (mh - k + y_1 - mx_1) / (\sqrt{m^2 + 1}) \right\} = D \quad (4.28)$$

Aquí:

D: es la distancia entre el eslabón L_2 y el centro $C(h, k)$ del objeto.

De acuerdo con la ecuación (4.28), se interpreta que la distancia "D" depende del valor de la pendiente "m" del eslabón L_2 y de las coordenadas del órgano efector (x_{OE}, y_{OE}) . El órgano efector se mueve sobre τ , lo cual significa que, para un cierto punto $P_2(x_2, y_2)$ sobre τ , existe una pendiente "m" tal que la ecuación (4.28) se cumpla para una magnitud de "D" igual a la del radio del objeto; es decir:

Si $D = r$, entonces el nuevo punto $P_2(x_2, y_2)$ determina la configuración límite del robot donde el eslabón L_2 es tangente al objeto con codo abajo, como se observa en la figura 4.22.

Por otro lado, al calcular "m" con:

$$m = \frac{(y_{OE} - y_U)}{(x_{OE} - x_U)} \quad (4.29)$$

y aplicando la ecuación (4.28), se obtiene el punto $P_1(x_1, y_1)$, el cual determina la configuración límite del robot donde el eslabón L_2 es tangente al objeto con codo arriba, como se observa en la figura 4.22.

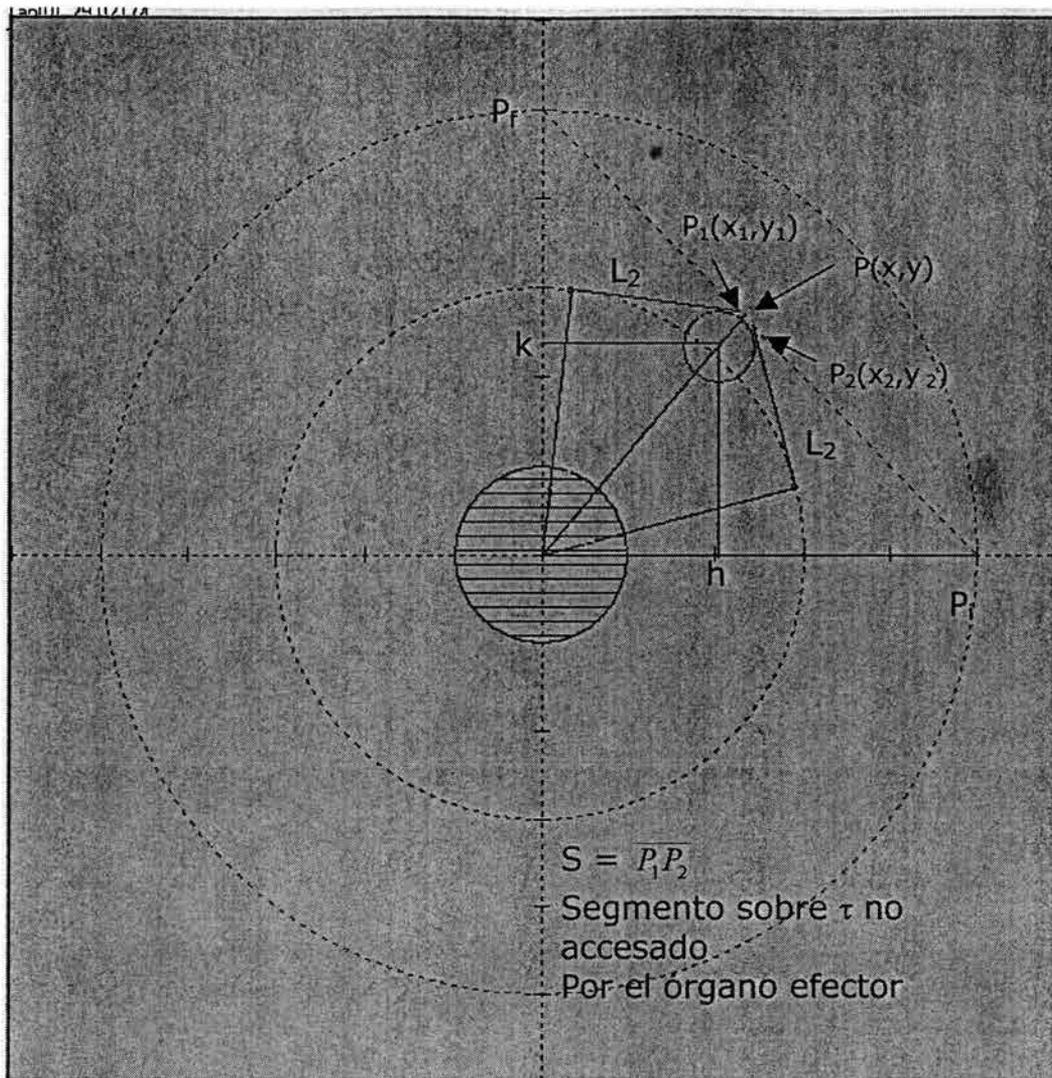


Figura 4.22 Secuencia de movimientos de L_2 para alcanzar la tangencia con el obstáculo

Las expresiones (4.27), (4.28) y (4.29) permiten determinar los puntos P_1 y P_2 sobre τ que definen la posición de tangencia entre R_o y O_b .

En este caso, la trayectoria τ no puede ser tocada totalmente por el órgano efector, y la nueva ruta de trabajo se recorre mediante los siguientes movimientos:

Movimientos asociados.- Según la figura 4.22 la nueva ruta del órgano efector del mecanismo será determinado por la siguiente secuencia:

- 1) De P_1 a P_2 en configuración de codo abajo con secuencias simultáneas.

- 2) De P_2 a P_i en configuración de codo abajo con secuencias simultáneas.
- 3) De P_i a P_1 en configuración de codo arriba con secuencias uno a uno y movimientos complementarios.
- 4) De P_1 a P_f en configuración de codo arriba con secuencias simultáneas.
- 5) De P_f a P_1 en configuración de codo arriba con secuencias simultáneas.
- 6) De P_1 a P_i en configuración de codo arriba con secuencias uno a uno.

Cabe señalar que con el uso de dichos movimientos, será posible evadir al obstáculo y que la ruta original τ no podrá ser recorrida totalmente por el órgano efector. La longitud del segmento S de τ no tocado por el órgano efector, será determinada por la distancia entre los puntos P_2 y P_3 .

Los tres criterios anteriores implican el uso de movimientos complementarios para mover al eslabón L_1 , esto es debido a que el objeto queda en un área de radio menor a la longitud de dicho elemento, en tanto que para el cuarto criterio, el eslabón L_1 no toca al objeto, en cuyo caso se establecerán algoritmos que no requieran movimientos de regreso para L_1 .

Criterio 4.- Si $h > h_2$ y $L_{R-r} > L_1$ y $L_{R+r} \leq R_v + r$, entonces la evasión ya no se hará a través de movimientos complementarios sino a través de una trayectoria paralela a la frontera del objeto. Por tanto, la trayectoria τ no podrá ser tocada en su totalidad y se deberá usar ambas configuraciones de codo.

En este caso, como se estableció anteriormente, la posición del objeto es tal que no toca el lugar geométrico de J_2 , y además, puede o no tocar a τ . Este caso será tratado especialmente en el siguiente capítulo.

Criterio 5.- Si $L_R - r > R_v$, entonces la posición del objeto no toca a τ y queda completamente fuera de A_{RR} por lo que el robot podrá ejecutar su operación normalmente.

Este es un caso singular, pues el objeto queda fuera de alcance del robot durante el recorrido del órgano efector por la ruta τ , por lo que las tareas pueden ser ejecutadas cíclicamente de una forma normal.

En el capítulo 5 se utilizará el criterio 4 para determinar las condiciones bajo las cuales el robot puede evadir al obstáculo a través de un segmento paralelo a su contorno, y, de la misma manera que para los criterios 2, 3 y 4, se determinarán las secuencias de movimientos, así como las configuraciones del multicuerpo para que el órgano efector recorra la nueva ruta de trabajo determinada.

CAPÍTULO V

Evasión a través de contorno expandido

Introducción. En el capítulo anterior, se establecieron las siguientes características principales :

- i) Las propiedades cinemáticas del robot R_o .
- ii) Los requerimientos de las tareas.
- iii) El área de trabajo (A_T) del robot.
- iv) El objeto O_b .

De acuerdo con la caracterización de cada uno de los anteriores elementos, se establecieron las siguientes relaciones:

- i) $\delta O_b \cap \delta A_{RR} = \emptyset$; objeto totalmente ajeno con el área de trabajo real reducida.
- ii) $O_b \subseteq A_{RR}$; objeto totalmente contenido en el área de trabajo real reducida.
- iii) $O_b \subset A_{RR}$; objeto parcialmente contenido en el área de trabajo real reducida.

Posteriormente se establecieron 5 criterios de evasión. Los criterios 1, 2, 3 y 5, fueron analizados con el objetivo de establecer las condiciones que permitan definir rutas alternas de trabajo para el robot evitando que colisione con el obstáculo.

En el presente capítulo se analizará el criterio 4, con el objetivo de definir una ruta alterna para el movimiento del órgano efector, evitando que cualquier parte del robot colisione con el obstáculo, la nueva ruta de evasión es paralela al contorno del obstáculo. Para lograr la evasión, se establecen los movimientos requeridos por el multicuerpo, así como las configuraciones de codo que debe adoptar.

Se determina también un punto sobre la ruta de evasión que indica el cambio de configuración del robot; es decir, para pasar de una configuración de codo abajo a una de codo arriba, y, al igual que en el capítulo anterior, se utilizan los conceptos de tangencia entre una recta y una circunferencia, así como el de distancia entre un punto y una recta.

En el capítulo VI, se desarrollarán los algoritmos derivados de los diferentes criterios de evasión, para la escritura de un programa en Visual Basic V6 que permita simular el movimiento y la capacidad de evasión del robot.

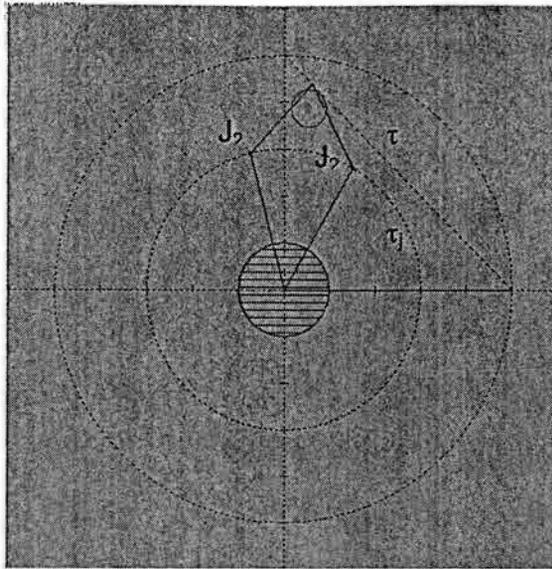
5.1 Condiciones de evasión

Criterio 4.- Si $h > h_2$ y $L_{R-r} > L_1$ y $L_{R+r} \leq R_v + r$, entonces la evasión ya no se hará a través de movimientos complementarios sino a través de una trayectoria paralela a la frontera del objeto. Por tanto, la trayectoria τ no podrá ser tocada en su totalidad y se deberá usar ambas configuraciones de codo.

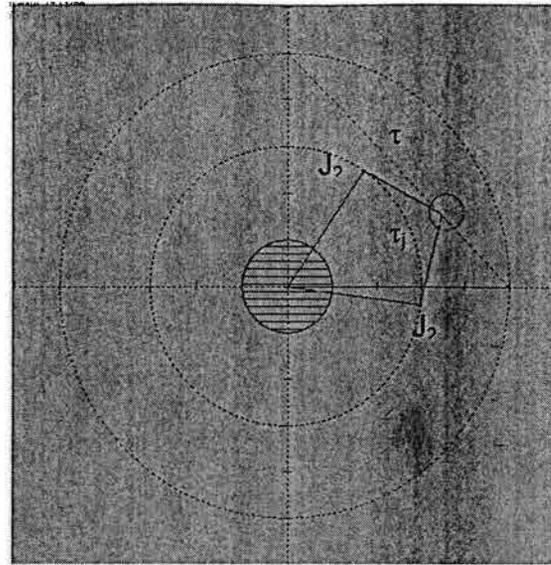
La posición del objeto es tal que:

- a) No interseca el lugar geométrico de J_2 .
- b) Puede, o no intersecar a τ .

En otras palabras, se tiene que, el objeto queda comprendido entre el lugar geométrico descrito por $J_2(\tau_j)$ y entre la trayectoria τ , llegando incluso a tocar a τ en algunos casos, según se muestra en la figura 5.1.



a) Obstáculo entre τ y τ_j



b) Obstáculo en intersección con τ

Figura 5.1 Posición canónica del robot respecto al objeto

En este caso, al igual que en los anteriores, puede efectuarse una evasión a través de movimientos complementarios, sin embargo, ahora existe la diferencia de que el eslabón L_1 no choca con el objeto, sino que es el eslabón L_2 o el órgano efector los que pueden originar la colisión. El interés entonces es el de lograr evadir al obstáculo a través de una trayectoria que sea paralela al contorno del objeto. Se puede observar que independientemente de los movimientos realizados por el manipulador, la trayectoria τ no podrá tocarse al 100% ya que L_2 tendrá al menos que adoptar una posición de tangencia con respecto al obstáculo, la cual difiere con la posición básica mostrada en la misma figura 5.1. La posición básica es la configuración que toma el robot cuando su órgano efector toca el punto común entre el segmento τ y la recta que parte del origen del sistemas de coordenadas cartesianas, pasando por el centro el obstáculo.

De acuerdo con la situación descrita en el párrafo anterior, para que el robot realice su tarea cubriendo la mayor trayectoria τ posible, será necesario determinar :

- 1) Una trayectoria (τ_2) libre de colisión que sea paralela al contorno del área del objeto que está en condición de colisión, la cual se muestra en la figura 5.2 .
- 2) La configuración del mecanismo para recorrer la nueva trayectoria τ_n desde el punto P_i hasta el punto P_f .

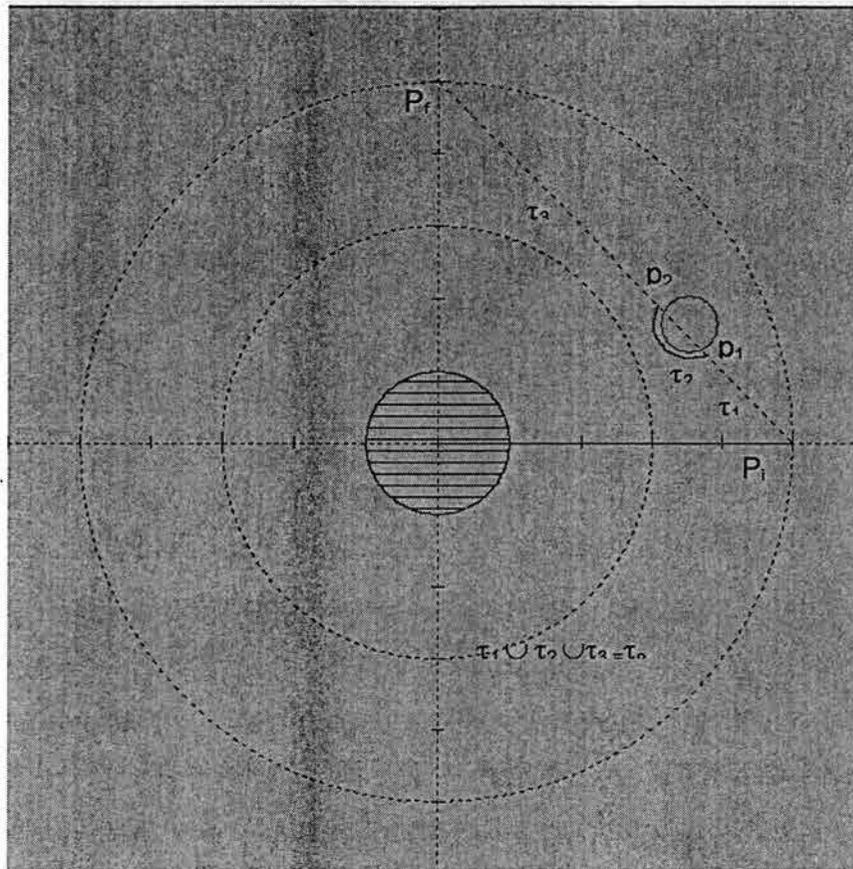


Figura 5.2 Nueva ruta de trabajo τ_n

5.2 Determinación de la trayectoria libre de colisión

De acuerdo con la figura 5.2, es posible obtener la trayectoria de evasión τ_2 , a través de la dilatación del área del objeto; es decir, si el objeto se representa por la ecuación de la circunferencia:

$$(x-k)^2 + (y-k)^2 = r^2 \quad (5.1)$$

entonces, su contorno es aumentado a través de un incremento en la longitud del radio, esto es:

$$(x-k)^2 + (y-k)^2 = (r + \Delta r)^2 \quad (5.2)$$

Donde Δr indica un incremento en la longitud del radio del objeto. Se considera que con el incremento Δr , el contorno toca la ruta de trabajo τ . De esta manera se obtendrá el nuevo segmento de trayectoria indicado como τ_3 (ver figura 5.2), la nueva trayectoria τ_n contiene los puntos : P_i , P_1 , P_2 y P_f y se describe como:

$$\tau_n = \tau_1 \cup \tau_2 \cup \tau_3$$

donde:

τ_n : representa la nueva trayectoria libre de obstáculos.

τ_1 : es una trayectoria rectilínea y pertenece a la trayectoria τ original.

τ_2 : es la nueva trayectoria paralela a un segmento del contorno del objeto.

τ_3 : es una trayectoria rectilínea y pertenece a la trayectoria τ original.

La nueva trayectoria τ_n es accesible al órgano efector del robot dado que se encuentra dentro del área de trabajo. Sin embargo, para ciertas posiciones del objeto, es posible que el eslabón L_2 colisione con el objeto aunque el órgano efector no lo haga, esto implicará la imposibilidad de seguir un contorno totalmente paralelo al objeto, en cuyo caso, será necesario incluir en la trayectoria de evasión un pequeño segmento rectilíneo.

La estrategia será entonces determinar una nueva trayectoria, mediante una expansión mínima arbitraria del contorno del objeto. Para ello, se debe establecer un incremento Δr mínimo y evaluar la distancia entre el eslabón L_2 y el objeto en

cada punto del nuevo recorrido tocado por el órgano efector, de tal manera, que si al detectar que la distancia entre el eslabón L_2 y el objeto sea menor que Δr , entonces modificar nuevamente la trayectoria anterior en forma rectilínea hasta alcanzar la trayectoria original, lo anteriormente dicho toma en cuenta que para ciertas posiciones del objeto:

- i) Será posible evadir el obstáculo a través de una nueva trayectoria totalmente paralela al objeto (parte del objeto), y
- ii) que en otras posiciones, la nueva trayectoria estará integrada por un recorrido paralelo al contorno del objeto y un segmento rectilíneo, que implica la menor distancia de L_2 al objeto. Las condiciones para ambos casos deben ser consideradas para la generación del algoritmo de control, ver figura 5.3.

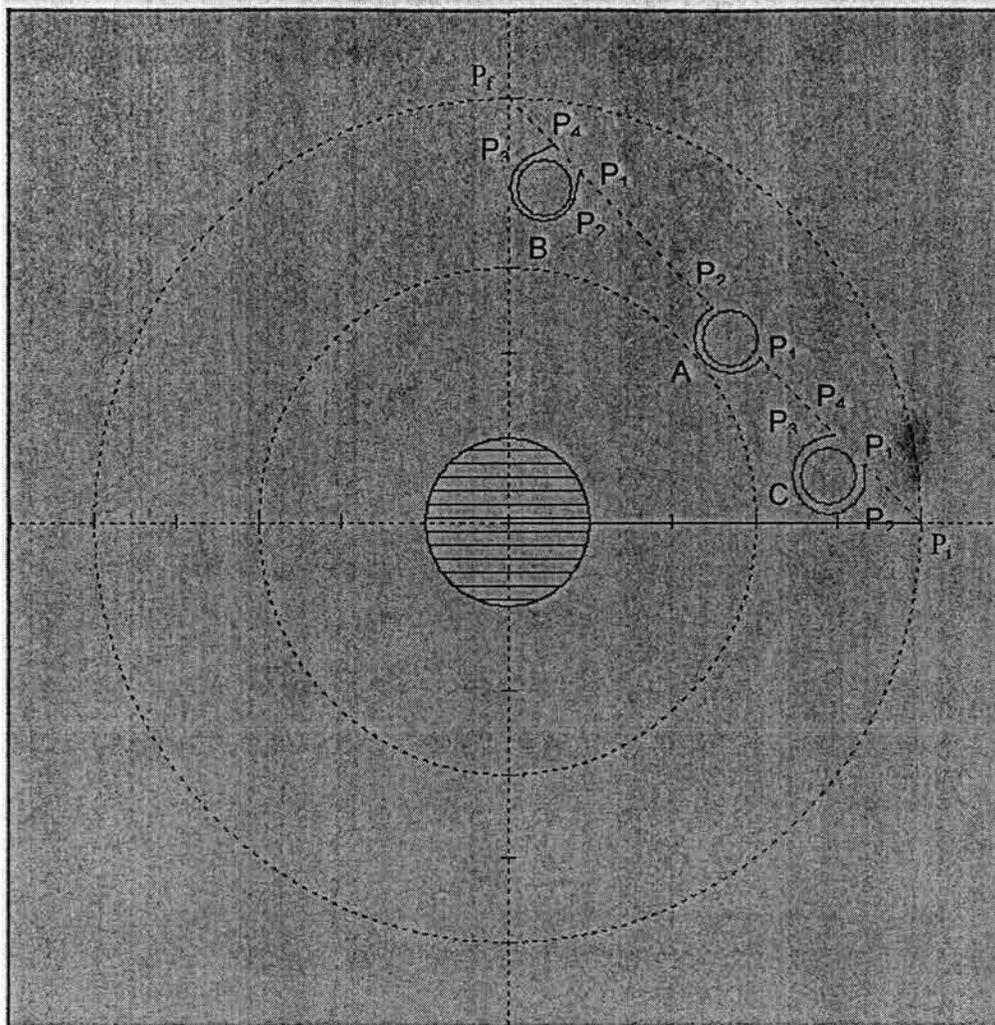


Figura 5.3 Posiciones relativas del objeto, según el criterio 5

En la posición B mostrada en la figura anterior, el órgano efector recorre la trayectoria indicada por los puntos P_1 , P_2 , P_3 y P_4 igual que en la posición C. En ambos casos la trayectoria de evasión es paralela al contorno del objeto en el segmento que une P_2 con P_3 y se observan dos segmentos rectilíneos que unen P_1 con P_2 y P_3 con P_4 , en tanto que en una posición del objeto como la correspondiente en A, el órgano efector puede recorrer la trayectoria de P_1 a P_2 totalmente paralela al contorno del objeto. En las tres posiciones se evita la colisión del eslabón L_2 con el objeto.

Tomando como base la anterior descripción, se puede caracterizar la trayectoria de evasión de la manera siguiente:

- 1) Posición A: Que toda la nueva trayectoria sea totalmente paralela al contorno.
- 2) Posición B y C: Que contenga segmentos rectilíneos, uno paralelo con el contorno.

5.2.1 Posición A

De acuerdo con la figura 5.3, se deben obtener las coordenadas de los puntos P_1 y P_2 pertenecientes a la trayectoria original τ y a la nueva trayectoria τ_n , por lo que la resolución del sistema formado por sus respectivas ecuaciones permitirá encontrar sus coordenadas.

Como se ha explicado en la sección 4.1.2, el órgano efector recorre la ruta de trabajo, partiendo del punto inicial $P_i(50,0)$ hasta el punto final $P_f(0,50)$ en forma rectilínea, por lo que la ecuación de la trayectoria τ es de la forma:

$$y - y_1 = m(x - x_1)$$

con:

$$m = \frac{(y_2 - y_1)}{(x_2 - x_1)}$$

siendo, $x_1 = y_1 = 0$ y $x_2 = y_2 = 50$

Por lo que $m = -1$ y la ecuación de τ es :

$$x + y = 50 \tag{5.2}$$

La nueva trayectoria τ_n , se obtiene al dilatar el contorno del objeto. La ecuación del contorno es de la forma:

$$(x - k)^2 + (y - k)^2 = r^2$$

Y la ecuación del contorno expandido será:

$$(x - k)^2 + (y - k)^2 = (r + \Delta r)^2 \quad (5.3)$$

Donde en ambos casos $C(h, k)$ es el centro del objeto, r es la longitud de su radio y Δr es el incremento arbitrario en la longitud del radio.

Así, se forma el sistema de dos ecuaciones con dos incógnitas con las expresiones (5.2) y (5.3), esto es:

$$\begin{aligned} x + y &= 50 \\ (x - k)^2 + (y - k)^2 &= (r + \Delta r)^2 \end{aligned}$$

Al sustituir la expresión (5.2) en la ecuación (5.3) y la expresión $(r + \Delta r)$ por r_i , se obtiene la expresión de segundo grado en "y" :

$$2y^2 + (2h - 2k - 100)y + (2500 - 100h + h^2 + k^2 - r_i^2) = 0$$

Al asignar:

$$a = 2, b = 2(h - k - 50) \text{ y } c = 2500 - 100h + h^2 + k^2 - r_i^2$$

se obtienen los valores de "x" e "y" aplicando:

$$a) y_1 = \left(-b + \sqrt{b^2 - 4ac} \right) / 2a \Rightarrow x_1 = 50 - y_1 \quad (5.4)$$

$$b) \quad y_2 = \left(-b - \sqrt{b^2 - 4ac}\right) / 2a \Rightarrow x_2 = 50 - y_2$$

$$c) \quad x_1 = 50 - y_1 \tag{5.5}$$

$$d) \quad x_2 = 50 - y_2$$

Así con las expresiones (5.4) y (5.5) se obtienen los puntos $P_1 (x_1, y_1)$ y $P_2 (x_2, y_2)$ de la nueva trayectoria y el órgano efector podrá evadir el objeto cubriendo el recorrido siguiente:

$$P_i \rightarrow P_1 \rightarrow P_2 \rightarrow P_f$$

Los segmentos $P_i \rightarrow P_1$ y $P_2 \rightarrow P_f$ corresponden a la trayectoria rectilínea original mientras que el segmento $P_1 \rightarrow P_2$ es el nuevo segmento de trayectoria paralela, generada a partir de la expansión del contorno del objeto en un incremento Δr arbitrario.

Como se observa en la figura 5.3, queda un segmento de la trayectoria original τ sin ser cubierta por el órgano efector, su longitud puede ser calculada de la forma siguiente:

$$\text{Longitud} = \overline{P_1 P_2}$$

O, equivalentemente,

$$\text{Longitud} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{5.6}$$

La variable Longitud, representa la distancia entre los puntos $P_1 (x_1, y_1)$ y $P_2 (x_2, y_2)$ e indica la magnitud del segmento de τ no tocado por el órgano efector.

5.2.2 Posición B y C

De acuerdo con las posiciones B y C mostradas en la figura 5.3, se observa que la nueva trayectoria de evasión puede contener uno o dos segmentos rectilíneos estando siempre presente el segmento paralelo al contorno, lo anterior dependerá de la configuración adoptada por el brazo, en otras palabras, es posible utilizar las configuraciones de codo arriba y codo abajo para eliminar alguno de los segmentos rectilíneos y convertirlo en un segmento también paralelo al contorno del objeto, dejando así solo un segmento no paralelo dentro de la nueva trayectoria. Esto será tratado en la sección relativa con las configuraciones del robot.

Retomando la figura 5.3, ahora deben ser determinados los puntos P_1 , P_2 , P_3 y P_4 . El siguiente análisis es igualmente válido para las posiciones en B y C.

Considerando el objeto de la posición B mostrado en la figura 5.4, la nueva trayectoria desde P_i hasta P_f se integra por los segmentos :

- 1) $P_i \rightarrow P_1$; segmento rectilíneo original.
- 2) $P_1 \rightarrow P_2$; segmento rectilíneo de evasión.
- 3) $P_2 \rightarrow P_3$; segmento de evasión paralelo al contorno del objeto.
- 4) $P_3 \rightarrow P_4$; segmento rectilíneo de evasión.
- 5) $P_4 \rightarrow P_f$; segmento rectilíneo original.

En la figura 5.4, se pueden observar los puntos P_o y P'_o que pertenecen a la intersección del segmento rectilíneo τ con el contorno expandido del obstáculo, dichos puntos, pueden ser obtenidos con las expresiones (5.4) y (5.5), el segmento que une a P_o con P'_o es inaccesible al órgano efector.

De acuerdo con los movimientos efectuados por el robot cuando el objeto se encuentra en la posición A, el primer movimiento del órgano efector debe ser

$P_i \rightarrow P_o$, sin embargo, se supone que antes de que el órgano efector llegue a P_o , el eslabón L_2 llega a su estado de tangencia con la nueva trayectoria del contorno expandido, por lo que cualquier mínimo desplazamiento del órgano efector hacia P_o implicará una aproximación del eslabón L_2 hacia la colisión, dependiendo del incremento Δr considerado. Esto se observa en la figura 5.4.

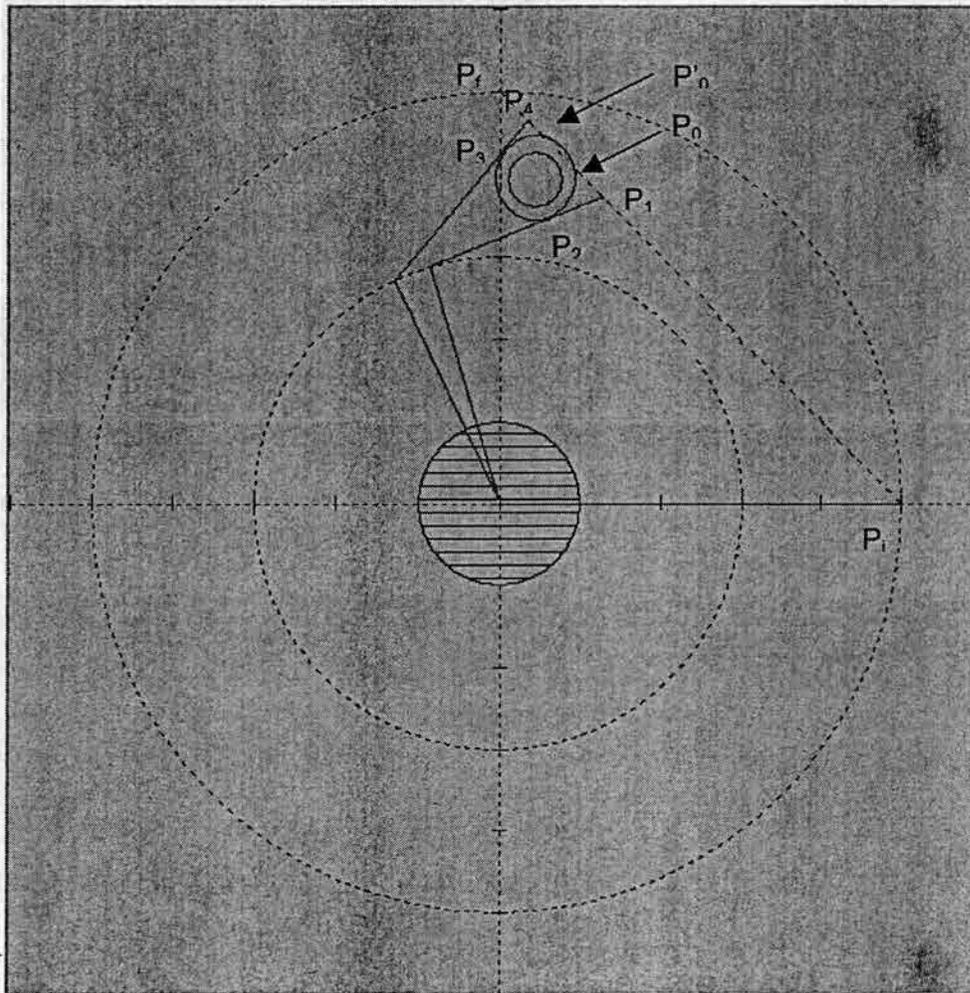


Figura 5.4 Posición de tangencia entre el eslabón L_2 y el objeto

5.2.3 Condiciones de tangencia entre el obstáculo y el contorno expandido

De acuerdo con la figura 5.4, se debe encontrar la ecuación de la recta tangente con el contorno expandido del obstáculo y que contiene al eslabón L_2 . Así, conociendo dicha ecuación, se podrán calcular las coordenadas de los puntos P_2 y P_1 .

Las coordenadas del punto P_2 se obtienen resolviendo el sistema con las ecuaciones (5.3) y (4.26) del contorno expandido y del eslabón L_2 respectivamente, y considerando la siguiente metodología:

- a) Mover el órgano efector sobre τ lo cual implica conocer las coordenadas de los puntos asociados con el órgano efector y J_2 que son los extremos del eslabón L_2 .
- b) Conociendo las coordenadas de $J_2(x,y)$ y el órgano efector (x',y') calcular la ecuación de la recta que contiene al eslabón L_2 .
- c) Calcular la distancia entre el $C(h, k)$ y el eslabón L_2 :
Si Distancia $> (r + \Delta r)$, entonces regresar al paso a).
Si Distancia $= (r + \Delta r)$, entonces solucionar el sistema formado por las ecuaciones del contorno expandido y el eslabón L_2 para encontrar P_2 . En esta condición las coordenadas de P_1 corresponden al punto de intersección entre la recta que contiene al eslabón L_2 con el segmento que representa a τ .

Al considerar los pasos mencionados anteriormente se tiene:

La ecuación de la recta tangente al contorno expandido del obstáculo que contiene al eslabón L_2 es de la forma:

$$y - y_1 = m(x - x_1)$$

o, equivalentemente:

$$mx - y + (y_1 - mx_1) = 0 \quad (5.7)$$

donde (x_1, y_1) son las coordenadas de $J_2(x, y)$ o del órgano efector (x', y') , y la pendiente m está dada por:

$$m = (y' - y)/(x' - x) \quad (5.8)$$

En la expresión (5.7) se observa que la pendiente m del eslabón L_2 depende de las coordenadas de $J_2(x, y)$ y de las coordenadas del órgano efector (x', y') , el objetivo es determinar la pendiente "m" cuando el eslabón L_2 es tangente al contorno expandido del objeto.

El valor de "m" puede ser calculado aplicando la expresión (4.6) de la distancia entre un punto y una recta, y considerando que en este caso, el punto es $C(h, k)$, la ecuación de la recta es de la forma de la expresión (5.7) y la distancia es representada por la expresión $r + \Delta r$, esto es:

$$(mh - k + (y_1 - mx_1)) / \pm \sqrt{m^2 + 1} = r + \Delta r \quad (5.9)$$

De la ecuación (5.9), se obtiene el valor de "m" y al sustituirse dicho valor en la expresión (5.7), se determina la ecuación de la recta que contiene al eslabón L_2 en posición tangente con el contorno expandido del obstáculo, y en consecuencia con el inciso c) de la metodología anteriormente indicada, se procede a calcular las coordenadas del punto P_2 , para lo cual se resuelve el sistema formado por la ecuación (5.7) y la (5.2); tal sistema es:

$$mx - y = mx_1 - y_1 \quad (5.10)$$

$$(x-h)^2 + (y-k)^2 = (r + \Delta r)^2$$

La solución de este sistema genera las coordenadas del punto P_2 mostradas en la figura 5.4.

Por otro lado, las coordenadas del punto P_1 se obtienen resolviendo el sistema de dos ecuaciones y dos incógnitas, formado por las expresiones (5.7) y (5.2) de las rectas que definen al eslabón L_2 y a τ ; el sistema es:

$$mx - y = mx_1 - y_1 \tag{5.7}$$

$$x + y = 50 \tag{5.2}$$

De las cuales se obtienen las coordenadas del punto $P_1(x, y)$:

$$y = (50m + y_1 - mx_1)/(m + 1)$$

$$x = 50 - y$$

El punto P_1 encontrado, indica la posición del órgano efector a partir del cual se mueve en línea recta hacia P_2 ; es decir, hasta aquí y con los resultados obtenidos, el órgano efector puede recorrer la siguiente trayectoria:

$$P_i \rightarrow P_1 \rightarrow P_2$$

A partir del punto P_2 el órgano efector recorrerá una trayectoria que es paralela al contorno del objeto, otra vez, sin embargo, se supone que antes de que el órgano efector llegue al punto P_o' , el eslabón L_2 se coloca en forma tangente con el contorno expandido del objeto, por lo que cualquier desplazamiento del órgano efector hacia P_o' acercará el eslabón L_2 hacia el objeto, y dependiendo de Δr la colisión será inminente, lo anterior se muestra en la figura 5.4.

Ahora, para encontrar las coordenadas de los puntos P_3 y P_4 , se aplica la metodología expuesta en la sección 5.1.3, esta es:

- a) Mover el órgano efector sobre τ_2 lo cual implica conocer las coordenadas del propio órgano efector y J_2 que son los extremos del eslabón L_2 .
- b) Conociendo las posiciones de $J_2(x,y)$ y el órgano efector (x',y') , calcular la ecuación de la recta que contiene al eslabón L_2 .
- c) Calcular la distancia entre el centro del objeto $C(h, k)$ y el eslabón L_2 :
- d) Si distancia $> (r + \Delta r)$, entonces, regresar al paso a).

Si Distancia = $(r + \Delta r)$, entonces solucionar el sistema formado por las ecuaciones del contorno expandido y el eslabón L_2 para encontrar P_3 . En esta condición las coordenadas de P_4 corresponden al punto de intersección entre la recta que contiene al eslabón L_2 con el segmento que representa a τ .

Igualmente, son aplicadas las ecuaciones (5.7), (5.8), (5.9), para resolver el sistema (5.10) y así determinar las coordenadas del punto P_3 , mientras que, resolviendo el sistema formado por las ecuaciones (5.7) y (5.2), se encuentran las coordenadas del puntos P_4 . Con el conocimiento de los puntos P_3 y P_4 , el órgano efector puede recorrer el tramo faltante:

$$P_2 \rightarrow P_3 \rightarrow P_4 \rightarrow P_f$$

Y, en consecuencia, conociendo los puntos: P_1 , P_2 , P_3 y P_4 el órgano efector recorre la siguiente trayectoria:

$$P_i \rightarrow P_1 \rightarrow P_2 \rightarrow P_3 \rightarrow P_4 \rightarrow P_f$$

logrando evadir el obstáculo.

Hasta aquí, se ha efectuado un análisis sobre la trayectoria libre de obstáculos. Sin embargo, la capacidad de que el robot pueda seguir la nuevas trayectorias definidas, depende de las características de su movimiento y configuraciones

adoptables. En la siguiente sección se realiza un análisis de las configuraciones admisibles.

5.3 Determinación de la configuración en la nueva trayectoria libre de colisión

En la figura 5.3, se observa que el eslabón L_1 puede girar libremente sin posibilidades de colisión con el objeto y, que el órgano efector del mecanismo puede recorrer la nueva trayectoria indicada tanto en su configuración de codo arriba como de codo abajo, siendo ahora L_2 quién puede colisionar con el obstáculo.

El criterio 4 indica que el objeto puede quedar localizado dentro del área especificada en la figura 5.5.

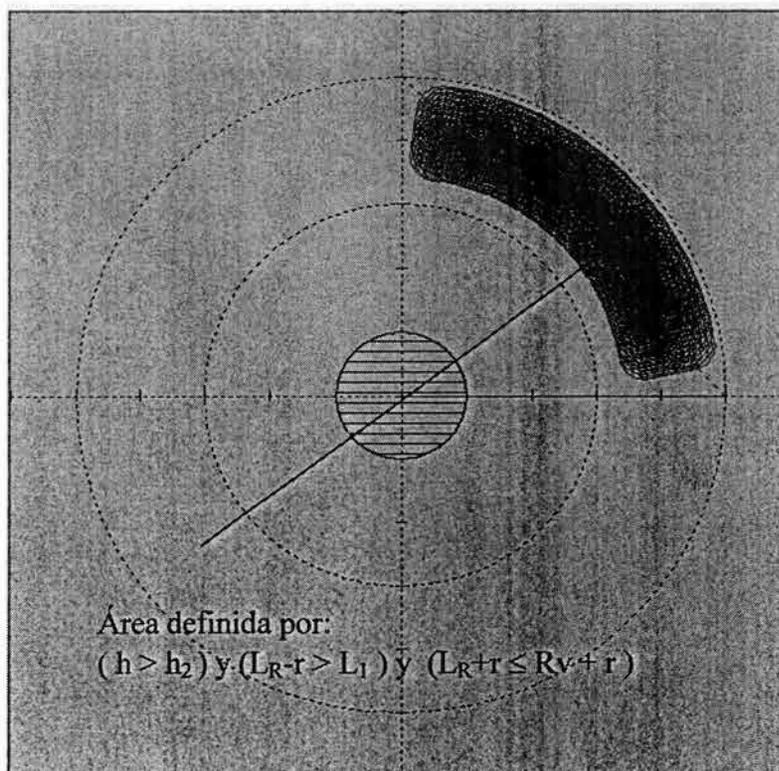


Figura 5.5 Región de presencia del obstáculo de acuerdo al criterio 4

En esta sección se determinará la configuración adoptada por el robot para recorrer la nueva trayectoria τ_n , considerando que:

- i) El objeto toca la trayectoria τ .
- ii) El objeto queda comprendido entre los lugares geométricos de J_2 (circunferencia con radio de longitud igual a la del segmento L_1) y τ , además la trayectoria τ solo es tocada por el contorno expandido.

En la figura 5.6 se observa el objeto en la posición A, (la cual fue mostrada anteriormente en la figura 5.3) y será tomada en cuenta para establecer conclusiones respecto a la configuración y movimiento del robot en su nueva ruta de evasión.

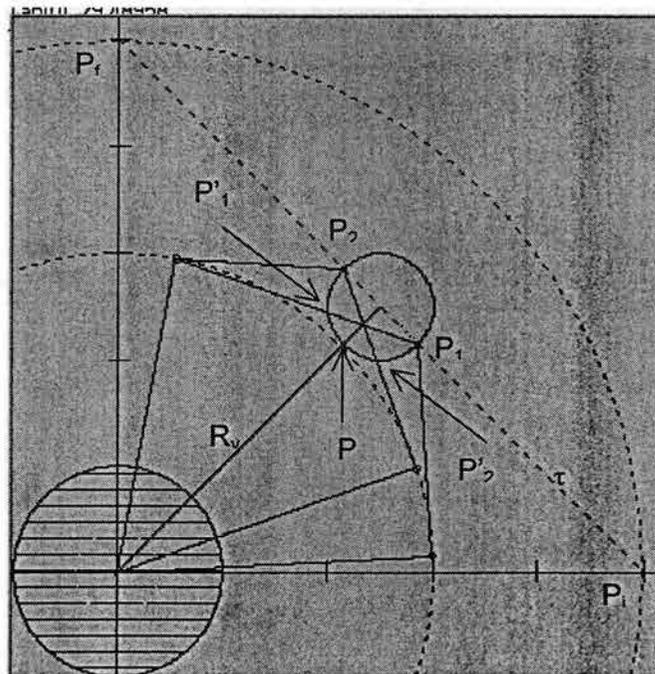


Figura 5.6 Configuración del robot en los puntos de intersección de τ con el contorno expandido

5.3.1 Seguimiento del contorno del objeto

En los dos siguientes apartados, se establecen las configuraciones del robot para que el órgano efector siga el contorno del objeto.

Al encontrar los puntos $P_1(x_1, y_1)$ y $P_2(x_2, y_2)$ de intersección entre τ y el contorno del objeto, es posible establecer las configuraciones de codo arriba y codo abajo en cada uno de ellos, verificándose las siguientes propiedades geométricas:

- 1) En el punto $P_1(x_1, y_1)$ la configuración de codo arriba contiene al eslabón L_2 en forma secante al objeto, tocándolo idealmente en los puntos P_1 y P'_1 , en tanto que la configuración de codo abajo solo toca al objeto en P_1 a través del órgano efector.
- 2) En el punto $P_2(x_2, y_2)$ la configuración de codo abajo contiene al eslabón L_2 en forma secante al objeto, tocándolo idealmente en los puntos P_2 y P'_2 , en tanto que la configuración de codo arriba solo toca al objeto en P_2 a través del órgano efector.
- 3) Si se toma en cuenta el eje R_v de simetría del objeto, mostrado en la figura 5.6, se observa que los puntos P_1 y P'_1 , P_2 y P'_2 se localizan separados por dicho eje y que el punto $P(x, y)$, es el lugar geométrico de intersección entre el radio vector R_v y el objeto.

5.3.1.1 Movimiento en configuración de codo abajo

En esta configuración el órgano efector es movido desde P_i hasta P_f a través de la siguiente nueva trayectoria:

$$P_i \rightarrow P_1 \rightarrow P_2 \rightarrow P_f$$

donde:

- i) $P_i \rightarrow P_1$; es un segmento rectilíneo que pertenece a la trayectoria τ original.
- ii) $P_1 \rightarrow P_2$; trayectoria que sigue el contorno del objeto.
- iii) $P_2 \rightarrow P_f$; es un segmento rectilíneo perteneciente a la trayectoria τ original.

Es posible observar que la trayectoria $P_1 \rightarrow P_2$ no podría ser cubierta por el sistema real, pues por la ley de impenetrabilidad de los cuerpos, no es posible que el robot adopte la configuración correspondiente al punto P_2 , dado a que ello implicaría que el robot R_0 y el obstáculo ocuparan el mismo espacio al mismo tiempo.

Si en la trayectoria indicada por $P_1 \rightarrow P_2$ se introduce un nuevo punto P , para obtener la nueva trayectoria :

$$P_1 \rightarrow P \rightarrow P_2$$

se puede hacer las siguiente observación:

En la trayectoria $P_1 \rightarrow P$, el eslabón L_2 no corta a R_v ; es decir, el órgano efector siempre está tocando los puntos del segmento $P_1 \rightarrow P$ evitando así el problema de impenetrabilidad que se daría en el sistema real en el punto P_2 .

5.3.1.2 Movimiento en configuración de codo arriba

En esta configuración el órgano efector es movido desde P_i hasta P_f a través de la siguiente nueva trayectoria:

$$P_i \rightarrow P_1 \rightarrow P_2 \rightarrow P_f$$

donde:

- i) $P_i \rightarrow P_1$ es un segmento rectilíneo perteneciente a la trayectoria τ original.
- ii) $P_1 \rightarrow P_2$ trayectoria que sigue e contorno del objeto.
- iii) $P_2 \rightarrow P_f$ es un segmento rectilíneo perteneciente a la trayectoria τ original.

Como en el caso analizado en la sección 5.2.1.1, la trayectoria $P_1 \rightarrow P_2$ no podría ser cubierta por el sistema real, por lo que se aplica la misma estrategia:

Si en la trayectoria indicada por $P_1 \rightarrow P_2$ se introduce un nuevo punto P , para obtener la nueva trayectoria :

$$P_1 \rightarrow P \rightarrow P_2$$

se puede hacer las siguiente observación:

En la trayectoria $P \rightarrow P_2$ el eslabón L_2 no corta a R_v ; es decir, el órgano efector siempre está tocando los puntos del segmento $P \rightarrow P_2$ evitando así el problema de impenetrabilidad que se daría en el sistema real en el punto P_1 .

De acuerdo con lo anterior, se puede concluir que, no es posible recorrer la trayectoria desde el punto P_1 hasta el punto P_2 utilizando una sola configuración del codo, sino que es necesario emplear ambas configuraciones. Sin embargo, será necesario todavía verificar que en los segmentos recorridos por cada configuración, no existan otras condiciones de tangencia entre R_o y el obstáculo, logrando con ello que el contorno del objeto sea tocado solamente por el órgano efector; es decir, se debe verificar que dentro de dichos segmentos, se satisfagan las siguientes condiciones:

- 1) En la configuración codo abajo, no exista un estado de tangencia entre el eslabón L_2 y el objeto, durante el recorrido del órgano efector del punto P_1 al punto P sobre el contorno expandido.
- 2) En la configuración codo arriba, no exista un estado de tangencia entre el eslabón L_2 y el objeto, durante el recorrido del órgano efector del punto P al punto P_2 sobre el contorno del objeto.

Las condiciones de tangencia mencionadas en el párrafo anterior fueron establecidas anteriormente en la sección 5.1.3.

Así se tiene que, para que el órgano efector siga la ruta de trabajo y el contorno del objeto desde P_i hasta P_f , mostrado en la figura 5.7, el robot tendrá que adoptar las siguientes configuraciones:

- 1) De P_i a P_1 en codo abajo.
- 2) De P_1 a P en codo abajo.
- 3) De P a P_2 en codo arriba.
- 4) De P_2 a P_f en codo arriba.

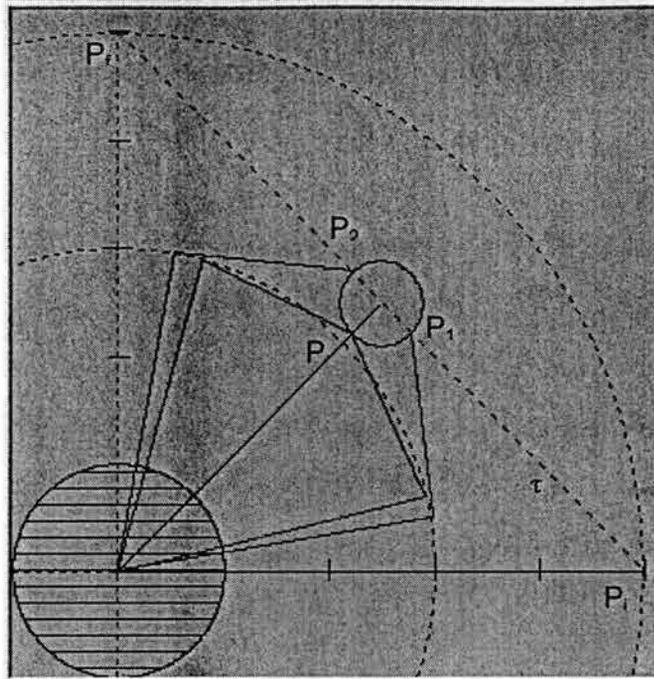


Figura 5.7 Cambio de configuración en el punto P

5.3.2 Configuración y secuencias para la trayectoria de contorno expandido

Para evitar el contacto entre el órgano efector y el objeto, se debe expandir el contorno de éste último como se ha explicado anteriormente, incrementando el radio del objeto en un cierto valor Δr arbitrario (puede ser utilizado el criterio de tangencia), en la figura 5.8 se muestra la nueva trayectoria correspondiente al contorno expandido del objeto.

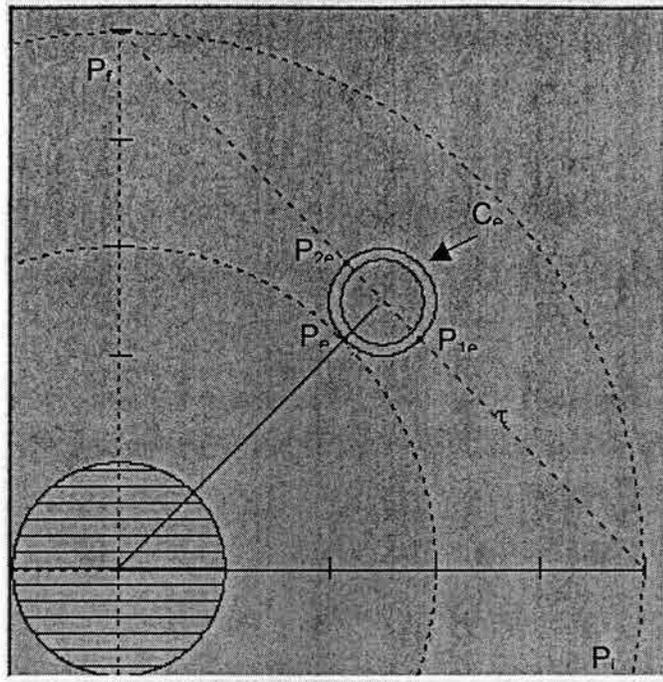


Figura 5.8 Intersección entre el contorno expandido y la ruta τ

De acuerdo con la figura anterior:

C_e es el contorno expandido dado por: $(x - k)^2 + (y - k)^2 = (r + \Delta r)^2$ y

P_e, P_{1e}, P_{2e} son puntos elementos de C_e .

La trayectoria está definida por:

- 1) $P_i \rightarrow P_{1e}$; configuración de codo abajo con movimientos simultáneos.
- 2) $P_{1e} \rightarrow P_e$; configuración de codo abajo con movimientos simultáneos.
- 3) $P_e \rightarrow P_e$; movimiento 1 a 1 con complementos.
- 4) $P_e \rightarrow P_{2e}$; configuración de codo arriba con movimientos simultáneos.
- 5) $P_{2e} \rightarrow P_f$; configuración de codo arriba con movimientos simultáneos.

5.3.3 Criterios para la determinación de Pe

Para establecer las coordenadas del punto $P_e(x_1, y_1)$ donde se efectúa el cambio de configuración del codo, se plantea el siguiente problema:

“ Conocidos: el centro $C(h, k)$, el radio r del obstáculo, el incremento Δr del radio y las restricciones del problema, determinar la ecuación del contorno expandido, radio vector (eje de simetría del obstáculo) y las coordenadas del punto $P_e(x_1, y_1)$ que indica el cambio de configuración del mecanismo en la nueva trayectoria definida por el contorno expandido ”

De acuerdo con el problema formulado anteriormente, la ecuación del contorno expandido es de la forma:

$$(x-h)^2 + (y-k)^2 = (r + \Delta r)^2 \quad (5.2)$$

La ecuación del segmento R_v que se define a partir del origen del sistema de coordenadas es :

$$y = mx \quad (5.11)$$

donde $m = k/h$

Por lo que, las coordenadas del punto P_e donde se da el cambio de configuración, se obtienen resolviendo el sistema formado por las dos ecuaciones (5.2) y (5.11).

Sustituyendo la expresión (5.11) en la ecuación (5.2) se obtiene:

$$(1 + m^2)x^2 - (2h + 2mk)x + h^2 + k^2 - r^2 = 0$$

Al considerar:

$$a = (1 + m^2)$$

$$b = -(2h + 2mk)$$

$$c = h^2 + k^2 - r^2$$

y usando:

$$x = \left(-b \pm \sqrt{b^2 - 4ac} \right) / 2a \quad (5.12)$$

se obtienen los valores de la abscisa "x", y al sustituirlas en la expresión (5.11) se determinan las ordenadas correspondientes, obteniendo así las coordenadas del punto $P_e(x,y)$.

En este capítulo se ha precisado lo siguiente :

- i) El contorno expandido.
- ii) La configuración adoptada por el mecanismo.
- iii) Secuencias de movimientos.
- iv) Condiciones de tangencia entre el eslabón L_2 y R_0 .

En el siguiente capítulo se describen los algoritmos que se derivan del conocimiento de las propiedades y condiciones descritas en cada uno de los anteriores incisos, así como de los criterios analizados en el capítulo IV. Estas condiciones son tomadas en cuenta para la formulación de algoritmos de planificación de la nueva trayectoria libre de obstáculos y para la simulación del movimiento del mecanismo.

CAPÍTULO VI

PROGRAMACIÓN Y SIMULACIÓN

Introducción.

En los capítulos anteriores, se han establecido relaciones, criterios y diferentes expresiones matemáticas para caracterizar:

- 1) La cinemática de posición de un robot de 2 GDL con juntas rotacionales, mediante la aplicación de operaciones de rotación y reflexión.
- 2) Las secuencias de movimientos asociados con los actuadores.
- 3) El sistema de trabajo (robot R_o , ruta de trabajo τ , objeto O_b)
- 4) Nuevas rutas de trabajo libres de obstáculos definidas bajo criterios geométricos.

En este capítulo se codifican los algoritmos resultantes de los diferentes criterios y modelos establecidos, relacionados con los cuatro incisos anteriores. Se diseñan diagramas de flujo y desarrollan programas, donde se prueba la validez de dichos modelos a través de un entorno gráfico de simulación.

Para el caso de los criterios establecidos en la caracterización de los movimientos del robot, el programa que se le asocia, presenta, en forma gráfica, el movimiento, las configuraciones adoptadas por el multicuerpo, así como el lugar geométrico generado por el órgano efector al pasar de una cierta configuración inicial a una final, igualmente, se presentan los valores de los parámetros de rotación.

En el módulo del programa correspondiente al problema de evasión de obstáculos, se presenta un entorno gráfico donde se caracterizan el área y la trayectoria de trabajo, se simula el movimiento del manipulador mediante un modelo de alambre

en un movimiento cíclico siguiendo la trayectoria de trabajo y se genera también la presencia de un obstáculo de posición aleatoria para mostrar la capacidad de evasión del robot al aplicar el criterio correspondiente.

6.1 Programación y simulación de secuencias para el robot Ro de 2 GDL

De acuerdo con el capítulo III sobre la caracterización de los movimientos de un robot de dos grados de libertad, tres tipos de movimientos fueron caracterizados: 1) uno a uno, 2) simultáneos y 3) híbridos. Estos movimientos se representaron por medio de tablas en código binario las cuales son consideradas para la escritura del programa de simulación correspondiente, en el propio capítulo III se muestran figuras producidas por el programa de simulación y que corresponden a los tres diferentes tipos de movimientos estudiados.

En la figura 6.1 se muestra la interfase del programa de simulación donde se reproducen los diferentes tipos de movimientos especificados en el capítulo III. El código asociado al módulo de este programa, puede ser consultado en el apéndice D.

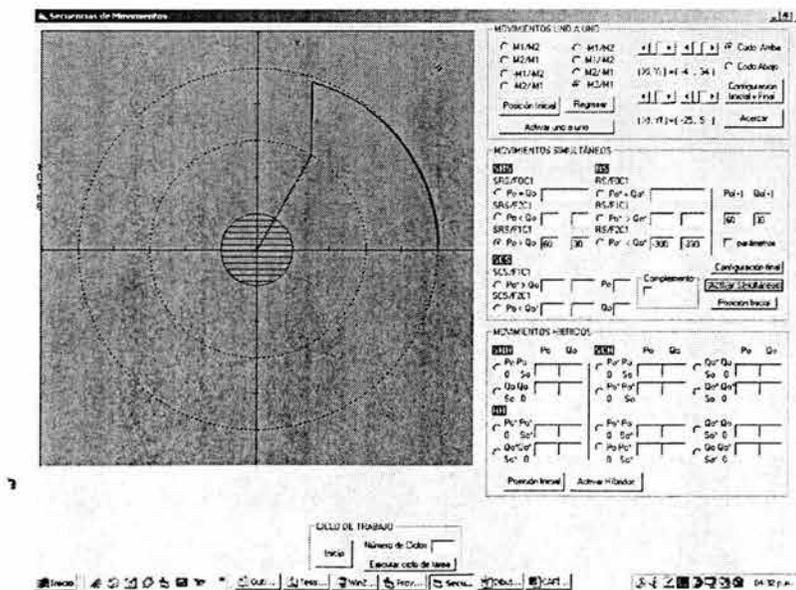


Figura 6.1 Interfase del módulo para la simulación del movimiento de Ro

6.2 Programación y simulación de los criterios de evasión

Los criterios de evasión se han definido a partir de la localización del obstáculo dentro del área de trabajo. En la figura 6.2 se indica dicha área de trabajo dividida en dos sectores:

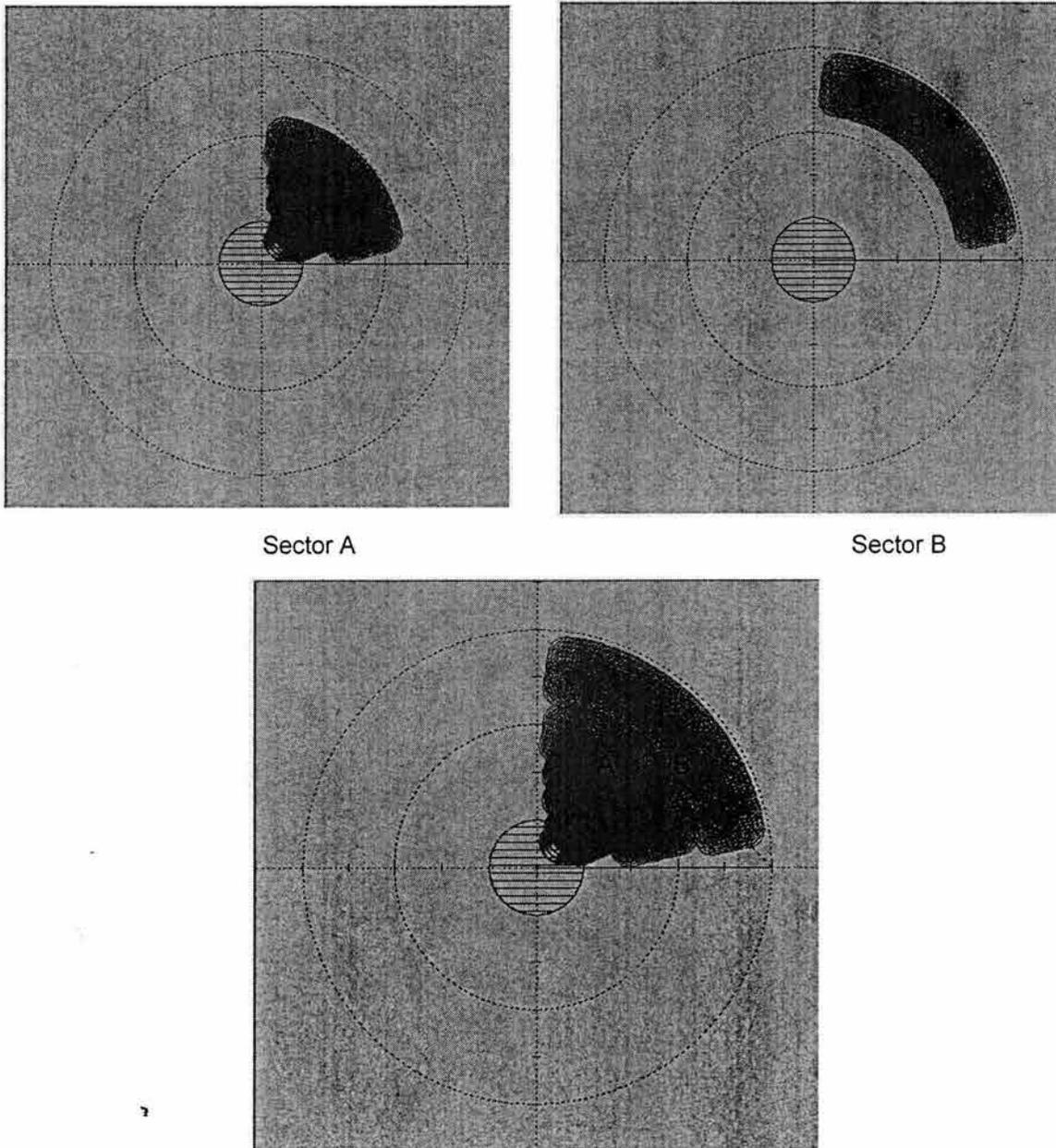


Figura 6.2 Área total de ubicación del obstáculo

De acuerdo al capítulo IV, se establecieron 3 condiciones que indican la posición del obstáculo en relación con el área de trabajo, estas son:

- i) $O_b \cap A_{RR} = \emptyset$; objeto totalmente ajeno al área de trabajo real reducida.
- ii) $O_b \subseteq A_{RR}$; objeto totalmente contenido en el área de trabajo real reducida.
- iii) $O_b \subset A_{RR}$; objeto parcialmente contenido en el área de trabajo real reducida.

Igualmente, de acuerdo con el conocimiento de las coordenadas del centro geométrico del objeto, se definieron los siguientes criterios de ubicación y evasión.

Criterio 1.- Si $h_1 < h < h_2$, entonces la trayectoria τ puede cubrirse totalmente utilizando ángulos complementarios y las configuraciones de codo arriba y codo abajo.

Aquí:

(h, k) : son las coordenadas del centro del objeto, en posición arbitraria.

(h_1, k_1) : son las coordenadas del centro del objeto en posición tangente al eslabón L_1 en ambas configuraciones de codo.

(h_2, k_2) : son las coordenadas del centro del objeto en posición tangente al eslabón L_2 en ambas configuraciones de codo.

Criterio 2.- Si $h < h_1$, entonces el eslabón L_1 debe ser movido hasta la posición de tangencia con O_b , por lo que quedará un segmento S sobre τ imposible de tocar. Para alcanzar la parte τ factible, se usaron movimientos complementarios, requiriendo además, ambas configuraciones. Cabe señalar que h y h_1 son las mismas del criterio 1.

Criterio 3.- Si $h > h_2$ y $L_R+r < R_v$, entonces L_2 debe ser movido hasta la posición de tangencia por lo que quedará un segmento de τ imposible de seguir, y la parte factible de alcanzar será a través de movimientos complementarios y aplicando las dos configuraciones del codo.

Aquí:

h y h_2 son las mismas del criterio 1.

L_R : es la longitud del segmento (OC) definido del origen del sistema cartesiano al centro del objeto.

r : es la longitud del radio del obstáculo.

En este caso la posición del objeto es tal que:

- a) Intersecta el lugar geométrico de las coordenadas de J_2 .
- b) No intersecta a τ .

Criterio 4.- Si $h > h_2$ y $L_R-r > L_1$ y $L_R+r \leq R_v + r$, entonces la evasión ya no se hará a través de movimientos complementarios sino a través de una trayectoria paralela a la frontera del objeto. Por tanto, la trayectoria τ no podrá ser tocada en su totalidad y se deberá usar ambas configuraciones de codo.

Aquí :

h y h_2 son las mismas del criterio 1.

L_R y r son los mismos que en el criterio 3.

R_v : segmento definido del origen del sistema de coordenadas, hasta τ y pasa por $C(h,k)$.

En este caso la posición del objeto es tal que:

- a) No intersecta el lugar geométrico de J_2 .
- b) intersecta a τ o no lo intersecta.

Criterio 5.- Si $L_R - r > R_v$, entonces la posición del objeto no toca a τ y queda completamente fuera de A_{RR} por lo que el robot podrá ejecutar su operación normalmente.

Aquí :

L_R , r y R_v son los mismos que en el criterio 3.

En la figura 6.3 se observa la interfase desarrollada en Visual Basic V6, implementada para el módulo correspondiente a la evasión y simulación de la tarea de trabajo.

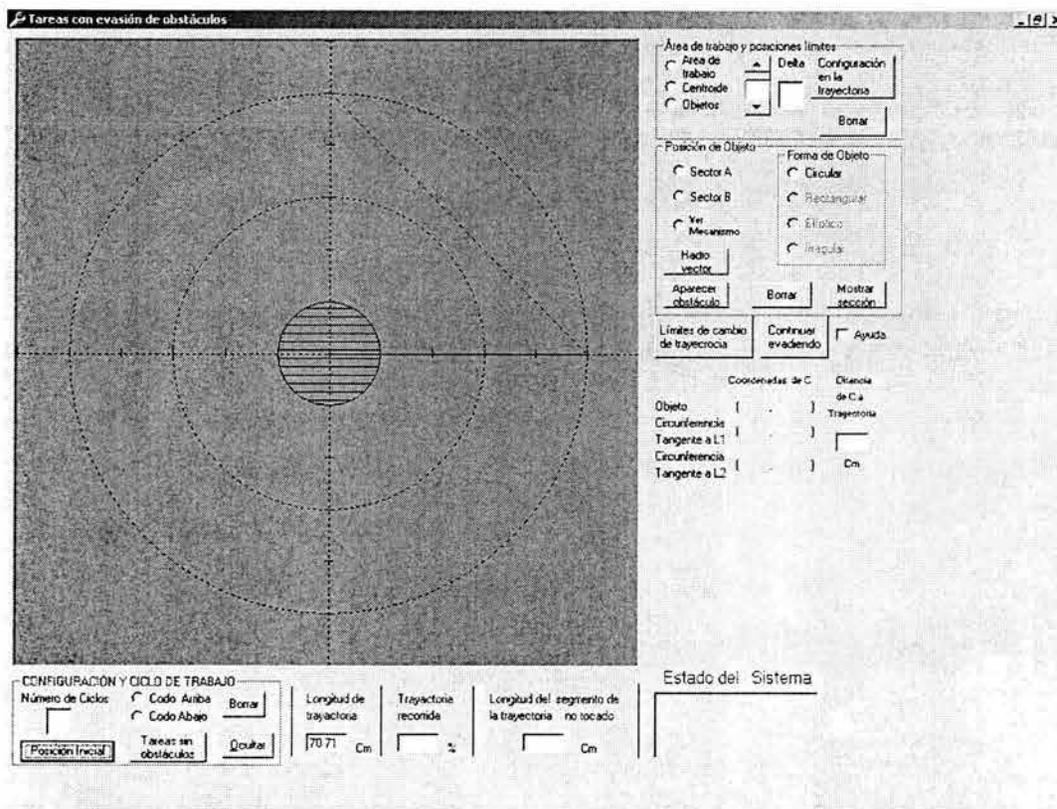


Figura 6.3 Interfase del programa de simulación para la evasión

Primeramente se presenta un diagrama de flujo general, donde se indica el uso de los anteriores criterios para determinar la secuencia del programa, y seguidamente

se describirá el diagrama y el código particulares implementados para cada uno de los bloques presentes.

El diagrama de flujo mostrado en la figura 6.4 considera los bloques mas generales, de acuerdo con cada uno de los criterios de evasión, considerando la posición del objeto, también se consideran dos bloque iniciales donde se genera la posición del objeto en forma aleatoria, así como la determinación de la posición de los eslabones L_1 y L_2 con respecto al objeto presente. Todo el movimiento del órgano efector se realiza de una manera cíclica desde la posición inicial en el punto P_i , hasta la posición final en el punto P_f .

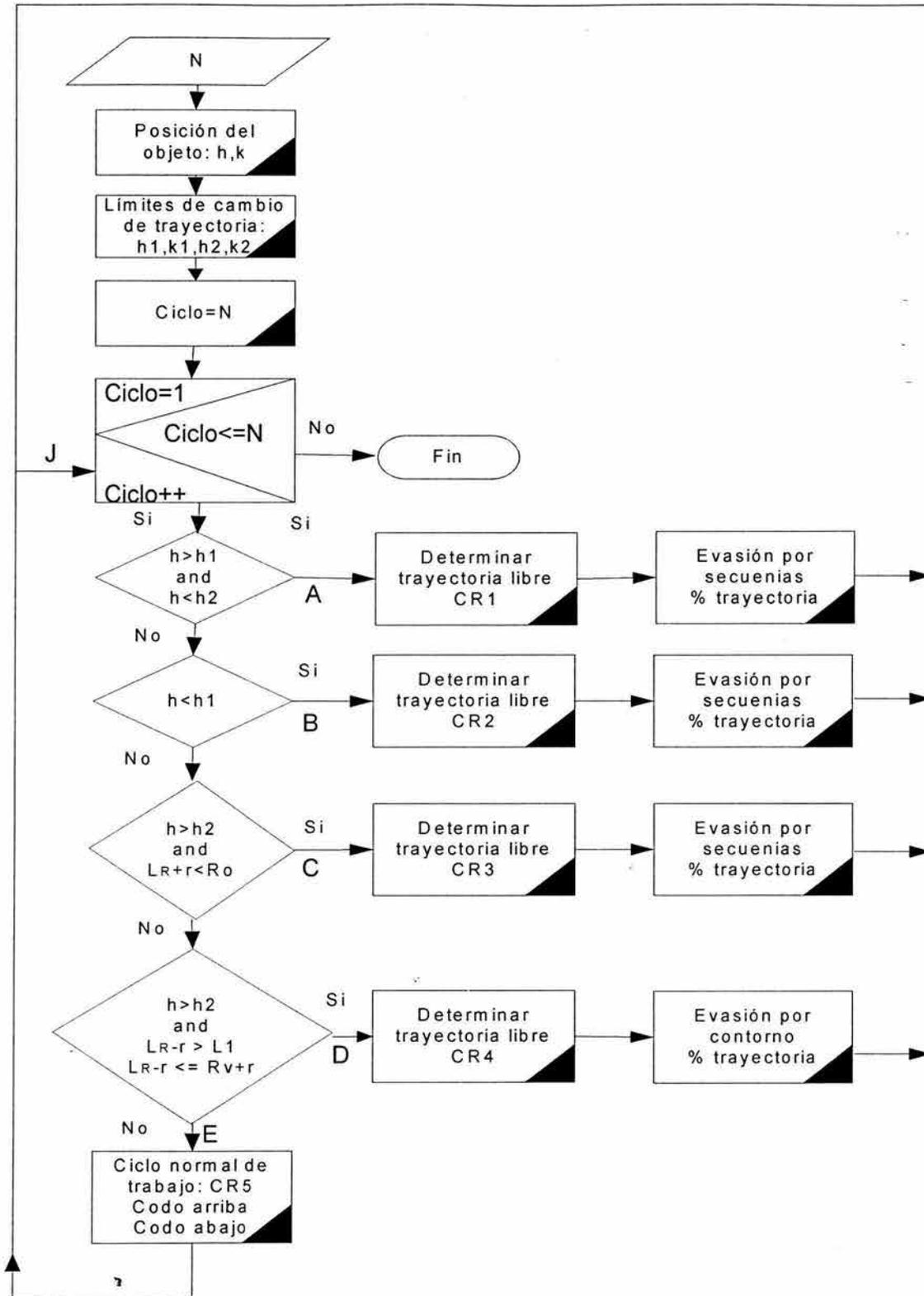


Figura 6.4 Diagrama de bloques del flujo general

A continuación se describe el código asociado a cada uno de los bloques del diagrama de flujo de la figura 6.4, el código está integrado por funciones asociadas con algún objeto de la interfase (principalmente, con botones de comando), y se expresa contenido entre las instrucciones:

Private Sub nombre del Objeto_Evento

End sub

Antes de mostrar el código correspondiente con cada criterio, se presentará un diagrama de bloques particular, también se explicará el propósito del mismo, y cuando sea necesario aclarar alguna línea o instrucción específica, se utilizará un comentario indicado entre corchetes “[]” o bien mediante el apóstrofe “ ’ ”, dentro del propio código.

6.2.1 Código asociado con el Bloque: Posición del objeto (h, k)

Para definir la ubicación del centro del objeto $C(h, k)$, primero se genera un vector con su punto inicial en el origen del sistema de coordenadas, con una magnitud aleatoria entre los valores de 1 cm y 50 cm y, de acuerdo a la magnitud se genera su dirección también en forma aleatoria entre los valores 1.5° y 72° . Éstos valores permiten definir las áreas A y B de la figura 6.1. Aquí se utiliza una función de números aleatorios proporcionada por el lenguaje VB expresada de la siguiente manera:

$\text{Int}((z * \text{Rnd}) + 1)$

Dicha función genera un valor aleatorio entre 1 y z, y para los propósitos de esta tesis, se ha modificado a la siguiente forma:

$\text{Int}((\text{Límite_superior} - \text{límite_inferior} + 1) * \text{Rnd} + \text{límite_inferior})$

para generar un valor aleatorio entre el valor del límite inferior y el valor del límite superior .

El siguiente código genera las coordenadas en forma aleatoria (h,k) del centro C del objeto, utilizando la forma de las coordenadas polares, la instrucción Option7.Value, Indica a la sección A del área de trabajo, en tanto que Option8.Value representa la sección B, mostradas en la figura 5.1.

Private Sub Command5_Click()

If Option7.Value = True Then

[En las variables magnitud y theta, se guardan los valores de la magnitud y dirección de un vector con un extremo anclado en el origen del sistema de coordenadas cartesianas, y con el segundo extremo definiendo las coordenadas del centro del obstáculo (h, k)]

```
magnitud = Int((35.85 - 5 - 7 + 1) * Rnd + 7)
If magnitud > 0 And magnitud < 5 Then
    theta = Int((38 - 36.55 + 1) * Rnd + 36.55)
Elseif magnitud > 5 And magnitud < 10 Then
    theta = Int((49 - 39.55 + 1) * Rnd + 39.55)
Elseif magnitud >= 10 And magnitud < 15 Then
    theta = Int((56 - 25.55 + 1) * Rnd + 25.55)
Elseif magnitud >= 15 And magnitud < 20 Then
    theta = Int((70 - 25.55 + 1) * Rnd + 25.55)
Elseif magnitud >= 20 And magnitud <= 25 Then
    theta = Int((75 - 12.5 + 1) * Rnd + 12.5)
Elseif magnitud >= 25 Then
    theta = Int((80 - 12.5 + 1) * Rnd + 12.5)
End If
```

```
h = magnitud * Cos(theta * 3.1416 / 180)
k = magnitud * Sin(theta * 3.1416 / 180)
```

```
Elseif Option8.Value = True Then
    theta = Int((82 - 10 + 1) * Rnd + 10) ' 7.29
    magnitud = Int((45 - 38 + 1) * Rnd + 38) '35
    h = magnitud * Cos(theta * 3.1416 / 180)
    k = magnitud * Sin(theta * 3.1416 / 180)
End If
```

[En las siguientes instrucciones, Rv guarda el valor de la longitud del vector desde el origen del sistema de coordenadas, hasta el centro (h,k), mientras que en longitud se guarda el valor de la distancia entre el centro c(h,k) y la recta de la trayectoria]

```
Rv = Sqr(h ^ 2 + k ^ 2)
longitud = (h + k - 50) / Sqr(2)
```

End Sub

6.2.2 Código asociado con el Bloque: Límites de cambio de trayectoria: h1, k1, h2, k2

Una vez conocido el centro $C(h,k)$ del objeto, se traza un segmento (radio vector) a partir del origen del sistema de coordenadas, que pase por $C(h,k)$ y toque a la trayectoria τ , se calcula este punto de intersección y se determinan las configuraciones de codo arriba y codo abajo, para luego establecer la posición de dos objetos tangentes a los eslabones L_1 y L_2 que indican la posición límite entre el robot R_0 y el propio objeto. Estos objetos contendrán la información de los parámetros (h_1, k_1) y (h_2, k_2) que posteriormente permitirán aplicar los criterios de evasión.

En el siguiente código se determinan las posiciones de los objetos tangentes con los eslabones del robot en las dos configuraciones de codo:

Private Sub Command6_Click()

[Hallar ecuación de radio vector]

$$y_r = (k / h) * x$$

[Hallar el punto de intersección entre la trayectoria del órgano efector y el radio vector]

$$x = (50 * h) / (k + h)$$

$$y = (50 * k) / (k + h)$$

[Hallar la configuración de codo arriba para el punto (x,y), aquí, P(0),P(1),q(0),q(1) son los parámetros del modelo cinemático inverso]

$$P(0) = (A * A^2 * x^2 + A * x^2 * (-B^2 + x^2 + y^2) - y * (\text{Sqr}(-A^2 * x^2 * (A^2 * A^2 + (-B^2 + x^2 + y^2)^2 - 2 * A^2 * (B^2 + x^2 + y^2)))))) / (2 * A^2 * x * (x^2 + y^2))$$

$$P(1) = (A * A^2 * y + A * y * (-B^2 + x^2 + y^2) + (\text{Sqr}(-A^2 * x^2 * (A^2 * A^2 + (-B^2 + x^2 + y^2)^2 - 2 * A^2 * (B^2 + x^2 + y^2)))))) / (2 * A^2 * (x^2 + y^2))$$

$$q(0) = (-A * A - B * B + x * x + y * y) / (2 * A * B)$$

$$q(1) = -(\text{Sqr}(-A^2 * x^2 * (A^2 * A^2 + (-B^2 + x^2 + y^2)^2 - 2 * A^2 * (B^2 + x^2 + y^2)))) / (2 * A^2 * B * x)$$

Call rota(P(), u1()) 'Llama a la función que realiza la operación de rotación

es1a(0) = A * Rpo(0) 'Coordenada x de codo arriba

es1a(1) = A * Rpo(1) 'Coordenada y del codo arriba

[En el siguiente If se determina el ángulo de cada elemento del manipulador, theta1_i es el ángulo del eslabón L1 y theta2_i es el ángulo del eslabón L2 en su posición de codo arriba]

If P(1) > 0 And P(0) > 0 Then

$$\text{theta1u} = \text{Atn}(P(1) / P(0)) * 180 / 3.1416$$

```

Elseif P(1) > 0 And P(0) < 0 Then
theta1u = (3.1416 + Atn(P(1) / P(0))) * 180 / 3.1416
Elseif P(1) < 0 And P(0) < 0 Then
theta1u = (3.1416 + Atn(P(1) / P(0))) * 180 / 3.1416
Elseif P(1) < 0 And P(0) > 0 Then
theta1u = ((2 * 3.1416) + Atn(P(1) / P(0))) * 180 / 3.1416
Elseif P(0) = 0 Then
theta1u = (90)
End If

```

```

If q(1) > 0 And q(0) > 0 Then
theta2u = Atn(q(1) / q(0)) * 180 / 3.1416
Elseif q(1) > 0 And q(0) < 0 Then
theta2u = (3.1416 + Atn(q(1) / q(0))) * 180 / 3.1416
Elseif q(1) < 0 And q(0) < 0 Then
theta2u = (3.1416 + Atn(q(1) / q(0))) * 180 / 3.1416
Elseif q(1) < 0 And q(0) > 0 Then
theta2u = ((2 * 3.1416) + Atn(q(1) / q(0))) * 180 / 3.1416
Elseif q(0) = 0 Then
theta2u = (90)
End If

```

```

theta1_j = theta1u
theta2_j = theta2u

```

```

Call rota(q(), Rpo())
pnd1(0) = es1a(0) + B * Rpo(0)
pnd1(1) = es1a(1) + B * Rpo(1)
longitud = Sqr(pnd1(0) ^ 2 + pnd1(1) ^ 2)

```

'abscisa del órgano efector
'ordenada del órgano efector

```

If (Rv > longitud + 4) Then
Option5.Value = 1
End If

```

'indica configuración de codo arriba

[Aquí pinta el brazo en el límite en codo arriba]

```

If Option12.Value = True Then
Picture1.Line (0, 0)-(es1a(0), es1a(1)), QBColor(9)
Picture1.Circle (es1a(0), es1a(1)), 0.3, QBColor(9)
Picture1.Line (es1a(0), es1a(1))-(pnd1(0), pnd1(1)), QBColor(9)
Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(RED)
End If

```

[Hallar la configuración de codo abajo para el punto (x,y), aquí, pp(0), pp(1), qp(0), qp(1) son los parámetros del modelo cinemático inverso]

```

pp(0) = (A * A ^ 2 * x ^ 2 + A * x ^ 2 * (-B ^ 2 + x ^ 2 + y ^ 2) + y * (Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2 + (-B ^ 2 + x ^ 2 + y ^ 2) ^ 2 - 2 * A ^ 2 * (B ^ 2 + x ^ 2 + y ^ 2)))) / (2 * A ^ 2 * x * (x ^ 2 + y ^ 2))
pp(1) = (A * A ^ 2 * y + A * y * (-B ^ 2 + x ^ 2 + y ^ 2) - (Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2 + (-B ^ 2 + x ^ 2 + y ^ 2) ^ 2 - 2 * A ^ 2 * (B ^ 2 + x ^ 2 + y ^ 2)))) / (2 * A ^ 2 * (x ^ 2 + y ^ 2))
qp(0) = (-A * A - B * B + x * x + y * y) / (2 * A * B)
qp(1) = (Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2 + (-B ^ 2 + x ^ 2 + y ^ 2) ^ 2 - 2 * A ^ 2 * (B ^ 2 + x ^ 2 + y ^ 2)))) / (2 * A ^ 2 * B * x)
Call rota(pp(), u1())
es1ap(0) = A * Rpo(0)

```

'Coordenada x del codo abajo

es1ap(1) = A * Rpo(1)

'Coordenada y del codo abajo

[En el siguiente If se determina el ángulo de cada elemento del mecanismo, theta1_f es el ángulo de L1 y theta2_f es el ángulo de L2 en su posición de codo abajo]

```
If pp(1) > 0 And pp(0) > 0 Then
  theta1u = Atn(pp(1) / pp(0)) * 180 / 3.1416
  Elseif pp(1) > 0 And pp(0) < 0 Then
  theta1u = (3.1416 + Atn(pp(1) / pp(0))) * 180 / 3.1416
  Elseif pp(1) < 0 And pp(0) < 0 Then
  theta1u = (3.1416 + Atn(pp(1) / pp(0))) * 180 / 3.1416
  Elseif pp(1) < 0 And pp(0) > 0 Then
  theta1u = ((2 * 3.1416) + Atn(pp(1) / pp(0))) * 180 / 3.1416
  Elseif pp(0) = 0 Then
  theta1u = (90)
End If
```

```
If qp(1) > 0 And qp(0) > 0 Then
  theta2u = Atn(qp(1) / qp(0)) * 180 / 3.1416
  Elseif qp(1) > 0 And qp(0) < 0 Then
  theta2u = (3.1416 + Atn(qp(1) / qp(0))) * 180 / 3.1416
  Elseif qp(1) < 0 And qp(0) < 0 Then
  theta2u = (3.1416 + Atn(qp(1) / qp(0))) * 180 / 3.1416
  Elseif qp(1) < 0 And qp(0) > 0 Then
  theta2u = ((2 * 3.1416) + Atn(qp(1) / qp(0))) * 180 / 3.1416
  Elseif qp(0) = 0 Then
  theta2u = (90)
End If
```

theta1_f = theta1u
theta2_f = theta2u

Call rota(qp(), Rpo())

pnd1p(0) = es1ap(0) + B * Rpo(0)

'abscisa del órgano efector

pnd1p(1) = es1ap(1) + B * Rpo(1)

'ordenada del órgano efector

[Aquí pinta el brazo en el límite, en codo abajo]

```
If Option12.Value = True Then
  Picture1.Line (0, 0)-(es1ap(0), es1ap(1)), QBColor(9)
  Picture1.Circle (es1ap(0), es1ap(1)), 0.3, QBColor(9)
  Picture1.Line (es1ap(0), es1ap(1))-(pnd1p(0), pnd1p(1)), QBColor(9)
  Picture1.Circle (pnd1p(0), pnd1p(1)), 0.1, QBColor(RED)
End If
```

[En la siguiente sección se encuentra el lugar geométrico del centro de las circunferencias que son tangentes a L1 y L2 en ambas configuraciones es decir C1(h1,k1) y C2(h2,k2) por lo tanto cualquier otra circunferencia fuera del rango establecido no permitirá cubrir toda la trayectoria] ³

[Inicio para circunferencias tangentes a L2]

```
If (pnd1p(0) - es1ap(0)) = 0 Then
  pnd1p(0) = pnd1p(0) + 0.0000001
```

End If

```
m1 = (pnd1(1) - es1a(1)) / (pnd1(0) - es1a(0))      'Pendiente de L2 en codo arriba
m2 = (pnd1p(1) - es1ap(1)) / (pnd1p(0) - es1ap(0)) 'Pendiente de L2 en codo abajo
m = (pnd1p(1) / pnd1p(0))
```

```
If pnd1(1) >= pnd1(0) Then
  h1 = (4 * Sqr(m1 ^ 2 + 1) + m1 * pnd1(0) - pnd1(1)) / (m1 - m)
  Elseif pnd1(1) < pnd1(0) Then
    h1 = (-4 * Sqr(m1 ^ 2 + 1) + m1 * pnd1(0) - pnd1(1)) / (m1 - m)
```

End If

```
If h1 >= pnd1(0) Then      ' Aquí se evita la circunferencia tangente sobre
  h1 = pnd1(0) - (h1 - pnd1(0)) ' la prolongación de L2
```

End If

k1 = (pnd1(1) / pnd1(0)) * h1

[Inicio para circunferencias tangentes a L1]

```
If es1ap(0) = 0 Then
  es1ap(0) = 0.000001
End If
If es1a(0) = 0 Then
  es1a(0) = 0.000001
End If
```

```
m2 = pnd1p(1) / pnd1p(0)
m1 = es1a(1) / es1a(0)
```

```
If pnd1(1) > pnd1(0) Then
  h2 = (4 * Sqr(m1 ^ 2 + 1)) / (m1 - m2)
  Elseif pnd1(1) <= pnd1(0) Then
    h2 = (-4 * Sqr(m1 ^ 2 + 1)) / (m1 - m2)
```

End If

```
If h2 < 0 Then      ' Aquí se evita la circunferencia tangente sobre
  h2 = -h2          ' la prolongación de L1
```

End If

k2 = m2 * h2

[Se indica la relación entre el centro del objeto, y el centro de los objetos en su posición límite]

```
If h > h2 And h < h1 Then
  Text9.Text = " Es posible la evasión usando complementos y además se cubre toda la
trayectoria "
  Elseif h < h2 Then
    Text9.Text = " Abrir L1 hasta ser tangente al objeto y usar complementos, pero no se cubre
toda la trayectoria "
  Else
    Text9.Text = " Abrir L2 hasta ser tangente al objeto y usar complementos, pero no se cubre
toda la trayectoria "
  End If
```


For m = 1 To ciclos

[va desde P_i hasta el punto $p(x,y)$ con movimientos simultáneos]

```
For x = 50 To xp Step -0.01
  If x = 0 Then
    x = 0.0000001
  End If
  y = -x + 50 ' ruta de trabajo
  Option6.Value = 1
  Call cin_inversa(x, y) 'se llama al modelo de la cinemática inversa
  Picture1.DrawStyle = 2
  Picture1.Line (0, 50)-(50, 0), QBColor(4)
  Picture1.DrawStyle = 0
  Call borrar ' se borra el modelo de alambre del robot
  Picture1.Circle (pnd1p(0), pnd1p(1)), 0.2, QBColor(1)
Next
```

For x = xp To 50 Step 0.01 'Aquí Ro regresa a P_i , con movimiento simultáneo.

```
  If x = 0 Then
    x = 0.0000001
  End If

  y = -x + 50 'Ruta de trabajo
  Call cin_inversa(x, y)
  Picture1.DrawStyle = 2
  Picture1.Line (0, 50)-(50, 0), QBColor(4)
  Picture1.DrawStyle = 0
  Call borrar ' se borra el modelo de alambre del robot
  Picture1.Circle (pnd1p(0), pnd1p(1)), 0.2, QBColor(1)
Next
```

[En el siguiente ciclo for, se mueve el eslabón L2, usando un movimiento secuencial, manteniendo fijo el eslabón L1]

```
For i = 0 To theta2_i - 360 Step -0.6
  Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(9)
  hp(0) = Cos(0 * 3.1416 / 180)
  hp(1) = Sin(i0 * 3.1416 / 180)
  gp(0) = Cos(i * 3.1416 / 180)
  gp(1) = Sin(i * 3.1416 / 180)
```

[Se llama a la función que aplica el modelo de la cinemática directa]

```
Call posición(hp(0), hp(1), gp(0), gp(1))
```

[La función pintar, dibuja la posición del robot, de acuerdo con los resultados del modelo de la cinemática directa]

```
Call pintar
Next
```

[En el siguiente ciclo for, se mueve el eslabón L1, usando movimiento secuencial, manteniendo fijo el eslabón L2, con ambos movimientos, el órgano efector se coloca de nuevo sobre el punto $P(x,y)$]

```

For i = 0 To theta1_i - 360 Step -0.6
  'Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(9)
  hp(0) = Cos(i * 3.1416 / 180)
  hp(1) = Sin(i * 3.1416 / 180)
  gp(0) = Cos(theta2_i * 3.1416 / 180)
  gp(1) = Sin(theta2_i * 3.1416 / 180)
  Call posición(hp(0), hp(1), gp(0), gp(1))
  Call pintar
Next

```

[Con el siguiente ciclo for, Ro continua con el resto de la trayectoria desde P(x,y) hasta Pi, con movimiento simultáneo]

```

Option5.Value = True
For x = xp To 0 Step -0.01
  If x = 0 Then
    x = 0.0000001
  End If
  y = -x + 50
  Option5.Value = 1
  Call cin_inversa(x, y)
  Picture1.DrawStyle = 2
  Picture1.Line (0, 50)-(50, 0), QBColor(4)
  Picture1.DrawStyle = 0
  Call borrar ' se borra el modelo de alambre del robot
  Picture1.Circle (x, y), 0.2, QBColor(1)
Next

```

[Con el siguiente for, Ro se mueve desde Pi hasta P(x,y) con movimiento simultáneo]

```

For x = 0 To xp Step 0.01
  If x = 0 Then
    x = 0.0000001
  End If
  y = -x + 50
  Call cin_inversa(x, y)
  Picture1.DrawStyle = 2
  Picture1.Line (0, 50)-(50, 0), QBColor(4)
  Picture1.DrawStyle = 0
  Call borrar ' se borra el modelo de alambre del robot
  Picture1.Circle (x, y), 0.2, QBColor(1)
Next

```

[En los siguiente dos ciclos for, Ro se mueve desde P(x,y) hasta Pi usando movimientos uno a uno en sentido contrario al del movimiento de ida de los dos primeros ciclos for]

```

For i = theta1_i - 360 To 0 Step 0.6
  'Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(9)
  hp(0) = Cos(i * 3.1416 / 180)
  hp(1) = Sin(i * 3.1416 / 180)
  gp(0) = Cos(theta2_i * 3.1416 / 180)
  gp(1) = Sin(theta2_i * 3.1416 / 180)
  Call posición(hp(0), hp(1), gp(0), gp(1))

```

```

Call pintar
'Picture1.Circle (x, y), 0.2, QBColor(1)
Next

```

```

For i = theta2_j - 360 To 0 Step 0.6
'Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(9)
hp(0) = Cos(0 * 3.1416 / 180)
hp(1) = Sin(i0 * 3.1416 / 180)
gp(0) = Cos(i * 3.1416 / 180)
gp(1) = Sin(i * 3.1416 / 180)
Call posición(hp(0), hp(1), gp(0), gp(1))
Call pintar
Next

```

```

Call FijarPosición 'se llama a una función que dibuja la posición final del sistema.
Option5.Value = 0
Option6.Value = 0

```

[pinta última posición]

```

Text11.Text = (100)
Picture1.Line (0, 0)-(es1a(0), es1a(1)), QBColor(9)
Picture1.Line (es1a(0), es1a(1))-(pnd1(0), pnd1(1)), QBColor(9)
Picture1.Circle (es1a(0), es1a(1)), 0.3, QBColor(9)
Picture1.Line (0, 0)-(es1ap(0), es1ap(1)), QBColor(9)
Picture1.Line (es1ap(0), es1ap(1))-(pnd1p(0), pnd1p(1)), QBColor(9)
Picture1.Circle (es1ap(0), es1ap(1)), 0.3, QBColor(9)
Next
Call home
Picture1.Circle (h, k), 4, QBColor(12)

```

[Si la condición $h < h_2$ es verdadera, entonces el eslabón L1 choca con el objeto, por tanto, este elemento debe moverse hasta ser tangente al objeto y luego evadir. Se calculan los puntos P3 y P4 mostrados en la figura 4.20, la ruta de trabajo no es tocada totalmente]

[Determinar trayectoria libre: CR2]

```

Elseif h < h2 Then ' Aquí el objeto choca con L1 por lo tanto, este elemento debe abrirse
' y moverse con complemento, aunque no podrá cubrir la trayectoria completa

```

```

Text9.Text = " Abrir L1 hasta ser tangente al objeto y usar complementos, pero no se cubre
toda la trayectoria "

```

[se determina la recta tangente (que contiene al eslabón L2) con el objeto, en las variables Aes1a(0), Aes1a(1), Aes1ap(0), y Aes1ap(1) se guardan coordenadas del codo en sus dos configuraciones]

```

r = 4
Ae = h ^ 2 - r ^ 2
Be = -2 * h * k
Ce = k ^ 2 - r ^ 2
m1 = (-Be + Sqr(Be ^ 2 - 4 * Ae * Ce)) / (2 * Ae)
m2 = (-Be - Sqr(Be ^ 2 - 4 * Ae * Ce)) / (2 * Ae)
Th1 = Atn(m1 + 10)
Th2 = Atn(m2 - 0.05)

```

```

Aes1a(0) = 30 * Cos(Th1)
Aes1a(1) = 30 * Sin(Th1)
Aes1ap(0) = 30 * Cos(Th2)
Aes1ap(1) = 30 * Sin(Th2)
signo = 1

```

[La función PuntoT recibe las coordenadas del codo abajo y determina las coordenadas del órgano efector mediante las expresión 4.24 y 4.25]

```
Call PuntoT(Aes1ap(0), Aes1ap(1), signo)
```

```

xp1 = pnd1(0): yp1 = pnd1(1)
ciclos = Int(Val(Text1))

```

```
For m = 1 To ciclos
```

[En el siguiente ciclo for, se mueve desde Pi hasta el punto P3, ver figura 4.20]

```

For x = 50 To xp1 Step -0.01
  If x = 0 Then
    x = 0.0000001
  End If
  y = -x + 50
  Option6.Value = 1
  Call cin_inversa(x, y)
  Picture1.DrawStyle = 2
  Picture1.Line (0, 50)-(50, 0), QBColor(4)
  Picture1.DrawStyle = 0
  Call borrar ' se borra el modelo de alambre del robot
  Picture1.Circle (pnd1p(0), pnd1p(1)), 0.2, QBColor(1)

```

```
Next
```

[En el siguiente ciclo, regresa a al punto inicial Pi]

```

For x = xp1 To 50 Step 0.01
  If x = 0 Then
    x = 0.0000001
  End If
  y = -x + 50
  Call cin_inversa(x, y)
  Picture1.DrawStyle = 2
  Picture1.Line (0, 50)-(50, 0), QBColor(4)
  Picture1.DrawStyle = 0
  Call borrar ' se borra el modelo de alambre del robot
  Picture1.Circle (pnd1p(0), pnd1p(1)), 0.2, QBColor(1)

```

```
Next
```

```
signo = -1
```

[La función PuntoT recibe las coordenadas del codo arriba y determina las coordenadas del órgano efector mediante las expresión 4.24 y 4.25]

```
Call PuntoT(Aes1a(0), Aes1a(1), signo)
```

```

xp2 = pnd1(0): yp2 = pnd1(1)
Option5.Value = 1
Vcontrol = 0
Call cin_inversa(xp2, yp2)
Vcontrol = 1

```

[En los siguientes dos ciclos for, Ro se mueve desde el punto P3 hasta el punto P4, ver figura 4.20]

```

For i = 0 To theta2_i - 360 Step -0.6
    hp(0) = Cos(0 * 3.1416 / 180)
    hp(1) = Sin(i0 * 3.1416 / 180)
    gp(0) = Cos(i * 3.1416 / 180)
    gp(1) = Sin(i * 3.1416 / 180)
    Call posición(hp(0), hp(1), gp(0), gp(1))
    Call pintar
Next

```

```

For i = 0 To theta1_i - 360 Step -0.6
    hp(0) = Cos(i * 3.1416 / 180)
    hp(1) = Sin(i * 3.1416 / 180)
    gp(0) = Cos(theta2_i * 3.1416 / 180)
    gp(1) = Sin(theta2_i * 3.1416 / 180)
    Call posición(hp(0), hp(1), gp(0), gp(1))
    Call pintar
Next

```

[Ro continua con el resto de la trayectoria desde el punto p3 al punto Pf]

```

Option5.Value = True
For x = xp2 To 0 Step -0.01
    If x = 0 Then
        x = 0.0000001
    End If
    y = -x + 50
    Option5.Value = 1
    Call cin_inversa(x, y)
    Picture1.DrawStyle = 2
    Picture1.Line (0, 50)-(50, 0), QBColor(4)
    Picture1.DrawStyle = 0
    Call borrar ' se borra el modelo de alambre del robot
    Picture1.Circle (x, y), 0.2, QBColor(1)
Next

```

[Ro continua desde el punto Pf al punto P3]

```

For x = 0 To xp2 Step 0.01
    If x = 0 Then
        x = 0.0000001
    End If
    y = -x + 50
    Call cin_inversa(x, y)
    Picture1.DrawStyle = 2
    Picture1.Line (0, 50)-(50, 0), QBColor(4)
    Picture1.DrawStyle = 0

```

```

Call borrar ' se borra el modelo de alambre del robot
Picture1.Circle (x, y), 0.2, QBColor(1)
Next

```

[En los dos siguientes ciclos for, Ro se regresa a Pi]

```

For i = theta1_i - 360 To 0 Step 0.6
'Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(9)
hp(0) = Cos(i * 3.1416 / 180)
hp(1) = Sin(i * 3.1416 / 180)
gp(0) = Cos(theta2_i * 3.1416 / 180)
gp(1) = Sin(theta2_i * 3.1416 / 180)
Call posición(hp(0), hp(1), gp(0), gp(1))
Call pintar
'Picture1.Circle (x, y), 0.2, QBColor(1)
Next

```

```

For i = theta2_i - 360 To 0 Step 0.6
'Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(9)
hp(0) = Cos(0 * 3.1416 / 180)
hp(1) = Sin(i0 * 3.1416 / 180)
gp(0) = Cos(i * 3.1416 / 180)
gp(1) = Sin(i * 3.1416 / 180)
Call posición(hp(0), hp(1), gp(0), gp(1))
Call pintar
Next

```

```

Call FijarPosición
Option5.Value = 0
Option6.Value = 0

```

[mantiene la última posición]

```

Picture1.Line (0, 0)-(es1a(0), es1a(1)), QBColor(9)
Picture1.Line (es1a(0), es1a(1))-(pnd1(0), pnd1(1)), QBColor(9)
Picture1.Circle (es1a(0), es1a(1)), 0.3, QBColor(9)
Picture1.Line (0, 0)-(es1ap(0), es1ap(1)), QBColor(9)
Picture1.Line (es1ap(0), es1ap(1))-(pnd1p(0), pnd1p(1)), QBColor(9)

```

```

Next
Call home
Picture1.Circle (h, k), 4, QBColor(12)

```

[Si no se cumple ninguna de las dos condiciones de los criterios 1 y 2, entonces se cumple la condición del criterio 3: $h > h_1$ y $L_R + r < R_V$, aquí se determinan los puntos P1 y P2 mostrados en la figura 4.22]

```

Else ' Aquí el objeto choca con L2 por lo tanto, este elemento debe abrirse
' y moverse con complemento, aunque no podrá cubrir la trayectoria completa

```

[Determinar trayectoria libre: CR3]

Text9.Text = " Abrir L2 hasta ser tangente al objeto y usar complementos, pero no se cubre toda la trayectoria "

xd = pnd1p(0): yd = pnd1p(1) 'Guarda coordenadas del punto p(x,y), ver figura 4.22

[En el siguiente ciclo for, se mueve el órgano efector sobre la ruta τ , y se guardan sus coordenadas xp1 y yp1, cuando el eslabón L2 es tangente con el obstáculo con codo abajo, así se obtiene el punto P2 mostrado en la figura 4.44]

```
For i = 0.01 To 5 Step 0.001
  A1 = 2
  B1 = 2 * xd - 2 * yd - 100
  C1 = xd ^ 2 + yd ^ 2 - 100 * xd + 2500 - i ^ 2
  divisor = 2 * A1
  raiz = Sqr(B1 ^ 2 - 4 * A1 * C1)
  Y1 = (-B1 - raiz) / divisor
  X1 = 50 - Y1
  xp1 = X1
  yp1 = Y1
  pnd1p(0) = X1
  pnd1p(1) = Y1
  Vcontrol = 0
  Call cin_inversa(X1, Y1)
  m = (pnd1p(1) - es1ap(1)) / (pnd1p(0) - es1ap(0))
  distancia = Abs((m * h - k + pnd1p(1) - m * pnd1p(0)) / Sqr(m ^ 2 + 1))
```

[Cuando la variable distancia toma el valor de 4, entonces se tiene la posición de tangencia entre el eslabón L2 y el objeto y se interrumpe el ciclo for]

```
If distancia = 4 Or (distancia > 4.001 And distancia < 4.01) Then
  Exit For
End If
Next
```

[El robot va de Pi hasta P2]

```
For x = 50 To xp1 Step -0.01
  If x = 0 Then
    x = 0.0000001
  End If
  y = -x + 50
  Option6.Value = 1 'empieza con CD
  Call cin_inversa(x, y)
  Picture1.DrawStyle = 2
  Picture1.Line (0, 50)-(50, 0), QBColor(4)
  Picture1.DrawStyle = 0
  Call borrar ' se borra el modelo de alambre del robot
  Picture1.Circle (pnd1p(0), pnd1p(1)), 0.2, QBColor(1)
Next
```

[En este for, el robot va desde el punto P2 hasta el punto Pi]

```
For x = xp1 To 50 Step 0.01
  If x = 0 Then
    x = 0.0000001
  End If
  y = -x + 50
  Call cin_inversa(x, y)
  Picture1.DrawStyle = 2
  Picture1.Line (0, 50)-(50, 0), QBColor(4)
  Picture1.DrawStyle = 0
  Call borrar ' se borra el modelo de alambre del robot
  Picture1.Circle (pnd1p(0), pnd1p(1)), 0.2, QBColor(1)
Next
```

[En el siguiente ciclo for, se mueve el órgano efector sobre a τ , y se guardan sus coordenadas xp2 y yp2, cuando el eslabón L2 es tangente con el obstáculo en codo arriba, así se obtiene el punto P1 mostrado en la figura 4.44]

```
For i = 0.01 To 5 Step 0.001
  A1 = 2
  B1 = 2 * xd - 2 * yd - 100
  C1 = xd ^ 2 + yd ^ 2 - 100 * xd + 2500 - i ^ 2
  divisor = 2 * A1
  raiz = Sqr(B1 ^ 2 - 4 * A1 * C1)
  Y2 = (-B1 + raiz) / divisor
  X2 = 50 - Y2
  xp2 = X2
  yp2 = 21
  pnd1(0) = X2
  pnd1(1) = Y2
  Vcontrol = 0
  Option5.Value = True
  Call cin_inversa(X2, Y2)
  m = (pnd1(1) - es1a(1)) / (pnd1(0) - es1a(0))
  distancia = Abs((m * h - k + pnd1(1) - m * pnd1(0)) / (Sqr(m ^ 2 + 1)))

  [Cuando la distancia es igual a 4, se tiene la posición de tangencia entre el eslabón L2 y el objeto y se interrumpe el ciclo for]

  If distancia = 4 Or (distancia > 4.001 And distancia < 4.01) Then
    Exit For
  End If
Next
```

[Con los dos siguientes ciclos for, el robot va desde el punto Pi hasta el punto P1]

```
För i = 0 To theta2_i - 360 Step -0.6
  hp(0) = Cos(0 * 3.1416 / 180)
  hp(1) = Sin(i0 * 3.1416 / 180)
  gp(0) = Cos(i * 3.1416 / 180)
  gp(1) = Sin(i * 3.1416 / 180)
  Call posición(hp(0), hp(1), gp(0), gp(1))
```

```

        Call pintar
Next
For i = 0 To theta1_i - 360 Step -0.6
    hp(0) = Cos(i * 3.1416 / 180)
    hp(1) = Sin(i * 3.1416 / 180)
    gp(0) = Cos(theta2_i * 3.1416 / 180)
    gp(1) = Sin(theta2_i * 3.1416 / 180)
    Call posición(hp(0), hp(1), gp(0), gp(1))
    Call pintar
Next

```

[En el siguiente ciclo el robot se mueve desde el punto P1 hasta el punto Pf]

```

Option5.Value = True
For x = xp2 To 0 Step -0.01
    If x = 0 Then
        x = 0.0000001
    End If
    y = -x + 50
    Option5.Value = 1
    Call cin_inversa(x, y)
    Picture1.DrawStyle = 2
    Picture1.Line (0, 50)-(50, 0), QBColor(4)
    Picture1.DrawStyle = 0
    Call borrar ' se borra el modelo de alambre del robot
    Picture1.Circle (x, y), 0.2, QBColor(1)
Next

```

[En el siguiente ciclo, Ro se mueve desde el punto Pf hasta el punto P1]

```

For x = 0 To xp2 Step 0.01
    If x = 0 Then
        x = 0.0000001
    End If
    y = -x + 50
    Call cin_inversa(x, y)
    Picture1.DrawStyle = 2
    Picture1.Line (0, 50)-(50, 0), QBColor(4)
    Picture1.DrawStyle = 0
    Call borrar ' se borra el modelo de alambre del robot
    Picture1.Circle (x, y), 0.2, QBColor(1)
Next

```

[En los dos siguientes ciclos for, Ro se mueve desde el punto P1 hasta el punto Pi]

```

For i = theta1_i - 360 To 0 Step 0.6
    hp(0) = Cos(i * 3.1416 / 180)
    hp(1) = Sin(i * 3.1416 / 180)
    gp(0) = Cos(theta2_i * 3.1416 / 180)
    gp(1) = Sin(theta2_i * 3.1416 / 180)
    Call posición(hp(0), hp(1), gp(0), gp(1))
    Call pintar
Next

```

```

For i = theta2_i - 360 To 0 Step 0.6

```

```

hp(0) = Cos(0 * 3.1416 / 180)
hp(1) = Sin(i0 * 3.1416 / 180)
gp(0) = Cos(i * 3.1416 / 180)
gp(1) = Sin(i * 3.1416 / 180)
Call posición(hp(0), hp(1), gp(0), gp(1))
Call pintar

```

Next

```

Call FijarPosición
Option5.Value = 0
Option6.Value = 0

```

[mantiene la última posición]

```

Picture1.Line (0, 0)-(es1a(0), es1a(1)), QBColor(9)
Picture1.Line (es1a(0), es1a(1))-(pnd1(0), pnd1(1)), QBColor(9)
Picture1.Circle (es1a(0), es1a(1)), 0.3, QBColor(9)
Picture1.Line (0, 0)-(es1ap(0), es1ap(1)), QBColor(9)
Picture1.Line (es1ap(0), es1ap(1))-(pnd1p(0), pnd1p(1)), QBColor(9)
Call home
Picture1.Circle (h, k), 4, QBColor(12)
End If

```

[El siguiente else se cumple cuando el objeto queda en la sección B mostrada en la figura 6.1, aquí se utiliza el criterio 4, con el objetivo de definir una ruta alterna para el movimiento del órgano efector, evitando que cualquier parte del robot, colisione con el obstáculo, la nueva ruta de evasión es paralela al contorno del obstáculo. Se determina también un punto sobre la ruta de evasión que indica el cambio de configuración del robot; es decir, para pasar de una configuración de codo abajo a una de codo arriba]

Else

[Determinar trayectoria libre: CR4]

[En estas líneas que le anteceden al ciclo for, se determinan los puntos de intersección (xp1,yp1) y (xp2,yp2) entre el contorno dilatado del obstáculo y la ruta de trabajo, dichos puntos se muestran en la figura 5.6 donde se indican como P₁ y P₂, se aplican las expresiones 5.4 y 5.5]

```

A1 = 2
B1 = 2 * h - 2 * k - 100
C1 = 2500 - 25 - 100 * h + h ^ 2 + k ^ 2 'quitó 64 y puse 16
divisor = 2 * A1
raiz = Sqr(B1 ^ 2 - 4 * A1 * C1)
y = (-B1 - raiz) / divisor
x = 50 - y
xp1 = x
yp1 = y

y = (-B1 + raiz) / divisor
x = 50 - y
xp2 = x
yp2 = y

```

[En el siguiente ciclo for el órgano efector recorre la ruta desde Pi hasta P1= Xp1, mostrada en la figura 5.6]

```

For x = 50 To xp1 Step -0.01
    If x = 0 Then
        x = 0.00001
    End If

    y = -x + 50

    Call cin_inversa(x, y)
    Picture1.DrawStyle = 2
    Picture1.Line (0, 50)-(50, 0), QBColor(4)
    Picture1.DrawStyle = 0
    Call borrar ' se borra el modelo de alambre del robot
    Picture1.Circle (pnd1p(0), pnd1p(1)), 0.2, QBColor(1)
Next

```

[Las siguientes 7 líneas, permiten determinar el punto Pe (xi,yi), donde ocurre el cambio de configuración del codo, se aplican las expresiones (5.11) y (5.12)]

```

A1 = (k / h) ^ 2 + 1
B1 = -2 * h - 2 * k * (k / h)
C1 = h ^ 2 + k ^ 2 - 25
divisor = 2 * A1
raiz = Sqr(B1 ^ 2 - 4 * A1 * C1)
xi = (-B1 - raiz) / divisor
yi = (k / h) * xi

```

[En el siguiente ciclo for, el órgano efector se mueve desde P_{1e} = xp1 hasta P_e=xi, la ruta seguida, se muestra en la figura 5.8]

```

For x = xp1 To xi Step -0.001
    If x = 0 Then
        x = 0.00001
    End If
    y = -Sqr(25 - (x - h) ^ 2) + k
    Call cin_inversa(x, y)
    Picture1.DrawStyle = 2
    Picture1.Line (0, 50)-(50, 0), QBColor(4)
    Picture1.DrawStyle = 0
    Call borrar ' se borra el modelo de alambre del robot
Next

```

```

Option5.Value = 1 'Define que la configuración sea de codo arriba
Call cin_inversa(xi, yi) 'Entra a cinemática inversa y calcula ángulos de eslabones L1 y L2

```

```

Option6.Value = 1 'Define que la configuración sea de codo abajo
Call cin_inversa(xi, yi) 'Entra a cinemática inversa y calcula ángulos de eslabones L1 y L2

```

```

If theta1_f > 270 Then
    theta1_f = -360 + theta1_f
End If

```



```

Picture1.DrawStyle = 2
Picture1.Line (0, 50)-(50, 0), QBColor(4)
Picture1.DrawStyle = 0
Call borrar ' se borra el modelo de alambre del robot
m = (pnd1(1) - es1a(1)) / (pnd1(0) - es1a(0))
distancia = (m * h - k + pnd1(1) - m * pnd1(0)) / (Sqr(m ^ 2 + 1))

```

[En el siguiente if se verifica que el eslabón L2 no colisione con el objeto; es decir, si la distancia entre el objeto y el eslabón L2 es tal que: $Distancia = r + \Delta r$, entonces se aplica la expresión (5.7) para que el órgano efector siga un trayecto rectilíneo, sin chocar con el objeto, según se muestra en la figura 5.3]

```

If (Abs(distancia) > 4.9 And Abs(distancia) < 5) Then
  ypp2 = (50 * m + pnd1(1) - m * pnd1(0)) / (m + 1)
  xpp2 = 50 - ypp2
  xe = pnd1(0): ye = pnd1(1)

  For xj = pnd1(0) To xpp2 Step 0.001
    y = m * xj + ye - m * xe
    Call cin_inversa(xj, y)
    Picture1.DrawStyle = 2
    Picture1.Line (0, 50)-(50, 0), QBColor(4)
    Picture1.DrawStyle = 0
    Call borrar ' se borra el modelo de alambre del robot
  Next

  xp2 = xpp2
  Exit For
End If

Next

```

[Pinta última posición]

```

Picture1.Line (0, 0)-(es1a(0), es1a(1)), QBColor(9)
Picture1.Line (es1a(0), es1a(1))-(pnd1(0), pnd1(1)), QBColor(9)
Picture1.Circle (es1a(0), es1a(1)), 0.3, QBColor(9)

```

[En este ciclo for, el órgano efector se mueve desde P2e hasta Pf, lo ruta se muestra en la figura 5.8]

```

For x = xp2 To 0 Step -0.01 'se va al punto expandido
  If x = 0 Then
    x = 0.00001
  End If
  y = -x + 50

  Call cin_inversa(x, y)
  Picture1.DrawStyle = 2
  Picture1.Line (0, 50)-(50, 0), QBColor(4)
  Picture1.DrawStyle = 0
  Call borrar ' se borra el modelo de alambre del robot
  Picture1.Circle (pnd1p(0), pnd1p(1)), 0.2, QBColor(1)
Next

```

[Pinta última posición]

```
Picture1.Line (0, 0)-(es1a(0), es1a(1)), QBColor(9)
Picture1.Line (es1a(0), es1a(1))-(pnd1(0), pnd1(1)), QBColor(9)
Picture1.Circle (es1a(0), es1a(1)), 0.3, QBColor(9)
```

```
End If
```

```
End If
```

```
End Sub
```

[En la siguiente sección del código se cumple que el objeto queda fuera del área real reducida de trabajo del robot, por lo tanto no existe interferencia, y la ruta de trabajo puede ser cubierta totalmente]

[Determinar trayectoria libre: CR5]

```
Else          ' aquí va código para cuando el objeto está fuera de alcance de L1 y L2
longitud = (h + k - 50) / Sqr(2)          'se calcula distancia entre la ruta y el Objeto.
If longitud >= 4 Then          ' no hay contacto entre el objeto y tao, por lo tanto no hay choque
    ciclos = Int(Val(Text1))          'Indica cuantos ciclos de trabajo se han programado.
For m = 1 To ciclos
```

[Inicia primer segmento desde Pi hasta Pf, con el órgano efetor sobre la ruta τ]

```
    For x = 50 To 0 Step -0.01
        If x = 0 Then
            x = 0.0000001
        End If

        y = -x + 50
        Call cin_inversa(x, y)
        Picture1.DrawStyle = 2
        Picture1.Line (0, 50)-(50, 0), QBColor(4)
        Picture1.DrawStyle = 0
        Call borrar          ' se borra el modelo de alambre del robot
```

```
    Next
```

[pinta última posición]

```
    If Option5.Value = True Then
        Picture1.Line (0, 0)-(es1a(0), es1a(1)), QBColor(9)
        Picture1.Line (es1a(0), es1a(1))-(pnd1(0), pnd1(1)), QBColor(9)
        Picture1.Circle (es1a(0), es1a(1)), 0.3, QBColor(9)
    ElseIf Option6.Value = True Then
        Picture1.Line (0, 0)-(es1ap(0), es1ap(1)), QBColor(9)
```

```

Picture1.Line (es1ap(0), es1ap(1))-(pnd1p(0), pnd1p(1)), QBColor(9)
Picture1.Circle (es1ap(0), es1ap(1)), 0.3, QBColor(9)
End If

```

```

For i = 0 To 10000           'Ciclo de retardo
For j = 0 To 1000
Next j
Next i

```

[Fin del primer segmento]

[inicia segundo segmento de regrese desde Pf hasta Pi]

```

For x = 0.00001 To 50 Step 0.01
If x = 50 Then
x = 49.999999
End If
y = -x + 50
Call cin_inversa(x, y)
Picture1.DrawStyle = 2
Picture1.Line (0, 50)-(50, 0), QBColor(4)
Picture1.DrawStyle = 0
Call borrar           ' se borra el modelo de alambre del robot
Next

```

[pinta última posición]

```

If Option5.Value = True Then
Picture1.Line (0, 0)-(es1a(0), es1a(1)), QBColor(9)
Picture1.Line (es1a(0), es1a(1))-(pnd1(0), pnd1(1)), QBColor(9)
Picture1.Circle (es1a(0), es1a(1)), 0.3, QBColor(9)
Elseif Option6.Value = True Then
Picture1.Line (0, 0)-(es1ap(0), es1ap(1)), QBColor(9)
Picture1.Line (es1ap(0), es1ap(1))-(pnd1p(0), pnd1p(1)), QBColor(9)
Picture1.Circle (es1ap(0), es1ap(1)), 0.3, QBColor(9)
End If

```

[Termina el segundo segmento]

Next m

6.2.4 Funciones implementadas

En el código hasta aquí visto, se utiliza la instrucción CALL, para llamar a una función que realiza alguna tarea específica, las funciones implementadas permiten la reducción en el número de líneas, a continuación se presenta el código de las funciones utilizadas.

Function cin_inversa(x As Single, y As Single)

[Ésta función recibe las coordenadas del órgano efector y determina los valores de los ángulos asociados a los eslabones L1 y L2 aplicando una rotación y una reflexión]

[P(0),P(1),q(0) y q(1) guardan los parámetros del modelo de la cinemática inversa en la configuración de codo arriba, aplicando la reflexión]

```

P(0) = -(A * A ^ 2 * x ^ 2 + A * x ^ 2 * (-B ^ 2 + x ^ 2 + y ^ 2) - y * (Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2
+ (-B ^ 2 + x ^ 2 + y ^ 2) ^ 2 - 2 * A ^ 2 * (B ^ 2 + x ^ 2 + y ^ 2)))))) / (2 * A ^ 2 * x * (x ^ 2 + y ^ 2))
P(1) = (A * A ^ 2 * y + A * y * (-B ^ 2 + x ^ 2 + y ^ 2) + (Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2 + (-B ^ 2 +
x ^ 2 + y ^ 2) ^ 2 - 2 * A ^ 2 * (B ^ 2 + x ^ 2 + y ^ 2)))))) / (2 * A ^ 2 * (x ^ 2 + y ^ 2))
coef = 1 / (2 * A ^ 2 * B * (x ^ 2 + y ^ 2) ^ 2)
S1 = A ^ 2 * A ^ 2 * A * (x ^ 2 - y ^ 2) - A * A ^ 2 * (x ^ 2 - y ^ 2) * (2 * B ^ 2 + x ^ 2 + y ^ 2)
S2 = A * B ^ 2 * (-x ^ 2 * x ^ 2 + y ^ 2 * y ^ 2 + B ^ 2 * (x ^ 2 - y ^ 2))
S3 = -2 * A ^ 2 * y * (Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2 + (-B ^ 2 + x ^ 2 + y ^ 2) ^ 2 - 2 * A ^ 2 * (B ^
2 + x ^ 2 + y ^ 2))))
S4 = 2 * B ^ 2 * y * (Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2 + (-B ^ 2 + x ^ 2 + y ^ 2) ^ 2 - 2 * A ^ 2 * (B ^
2 + x ^ 2 + y ^ 2))))
q(0) = (coef) * (S1 + S2 + S3 + S4)
q(1) = (1 / (2 * A ^ 2 * B * x * (x ^ 2 + y ^ 2) ^ 2)) * (-2 * A ^ 2 * A ^ 2 * A * x ^ 2 * y + 2 * A * B ^ 2 *
x ^ 2 * y * (-B ^ 2 + x ^ 2 + y ^ 2) + 2 * A ^ 2 * A * x ^ 2 * y * (2 * B ^ 2 + x ^ 2 + y ^ 2) + B ^ 2 * (x ^ 2
- y ^ 2) * (Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2 + (-B ^ 2 + x ^ 2 + y ^ 2) ^ 2 -
2 * A ^ 2 * (B ^ 2 + x ^ 2 + y ^ 2)))) + A ^ 2 * (-x ^ 2 + y ^ 2) * (Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2 + (-
B ^ 2 + x ^ 2 + y ^ 2) ^ 2 - 2 * A ^ 2 * (B ^ 2 + x ^ 2 + y ^ 2))))))

```

```

Call reflejar(P(), u1())
es1a(0) = A * Rpo(0)
es1a(1) = A * Rpo(1)

```

'Llama a la función que calcula una reflexión
'Coordenadas del codo

[La siguiente estructura condicional, permite determinar el valor del ángulo asociado a cada eslabón, utilizando los parámetros p1(), y q1()]

```

p1(0) = (A * A ^ 2 * x ^ 2 + A * x ^ 2 * (-B ^ 2 + x ^ 2 + y ^ 2) - y * (Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2 +
(-B ^ 2 + x ^ 2 + y ^ 2) ^ 2 - 2 * A ^ 2 * (B ^ 2 + x ^ 2 + y ^ 2)))))) / (2 * A ^ 2 * x * (x ^ 2 + y ^ 2))
p1(1) = (A * A ^ 2 * y + A * y * (-B ^ 2 + x ^ 2 + y ^ 2) + (Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2 + (-B ^ 2 +
x ^ 2 + y ^ 2) ^ 2 - 2 * A ^ 2 * (B ^ 2 + x ^ 2 + y ^ 2)))))) / (2 * A ^ 2 * (x ^ 2 + y ^ 2))
q1(0) = (-A * A - B * B + x * x + y * y) / (2 * A * B)
q1(1) = -(Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2 + (-B ^ 2 + x ^ 2 + y ^ 2) ^ 2 - 2 * A ^ 2 * (B ^ 2 + x ^ 2 +
y ^ 2)))) / (2 * A ^ 2 * B * x)

```

```

If p1(1) > 0 And p1(0) > 0 Then
  theta1u = Atn(p1(1) / p1(0)) * 180 / 3.1416
  Elseif p1(1) > 0 And p1(0) < 0 Then
    theta1u = (3.1416 + Atn(p1(1) / p1(0))) * 180 / 3.1416
  Elseif p1(1) < 0 And p1(0) < 0 Then
    theta1u = (3.1416 + Atn(p1(1) / p1(0))) * 180 / 3.1416
  Elseif p1(1) < 0 And p1(0) > 0 Then
    theta1u = ((2 * 3.1416) + Atn(p1(1) / p1(0))) * 180 / 3.1416
  Elseif p1(0) = 0 Then
    theta1u = (90)

```

End If

```

If q1(1) > 0 And q1(0) > 0 Then
  theta2u = Atn(q1(1) / q1(0)) * 180 / 3.1416
  Elseif q1(1) > 0 And q1(0) < 0 Then
    theta2u = (3.1416 + Atn(q1(1) / q1(0))) * 180 / 3.1416
  Elseif q1(1) < 0 And q1(0) < 0 Then
    theta2u = (3.1416 + Atn(q1(1) / q1(0))) * 180 / 3.1416
  Elseif q1(1) < 0 And q1(0) > 0 Then
    theta2u = ((2 * 3.1416) + Atn(q1(1) / q1(0))) * 180 / 3.1416
  Elseif q1(0) = 0 Then
    theta2u = (90)

```

End If

```

theta1_i = theta1u           ' ángulo del eslabón L1
theta2_i = theta2u         ' ángulo del eslabón L2

```

```

Call reflejar(q(), Rpo())
pnd1(0) = es1a(0) + B * Rpo(0)           'Coordenadas del órgano efector
pnd1(1) = es1a(1) + B * Rpo(1)

```

[pp(0),pp(1),qp(0) y qp(1) guardan los parámetros del modelo de la cinemática inversa en la configuración de codo abajo, usando una rotación]

```

Else
  pp(0) = (A * A ^ 2 * x ^ 2 + A * x ^ 2 * (-B ^ 2 + x ^ 2 + y ^ 2) + y * (Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2
+ (-B ^ 2 + x ^ 2 + y ^ 2) ^ 2 - 2 * A ^ 2 * (B ^ 2 + x ^ 2 + y ^ 2)))) / (2 * A ^ 2 * x * (x ^ 2 + y ^ 2))
  pp(1) = (A * A ^ 2 * y + A * y * (-B ^ 2 + x ^ 2 + y ^ 2) - (Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2 + (-B ^ 2 +
x ^ 2 + y ^ 2) ^ 2 - 2 * A ^ 2 * (B ^ 2 + x ^ 2 + y ^ 2)))) / (2 * A ^ 2 * (x ^ 2 + y ^ 2))
  qp(0) = (-A * A - B * B + x * x + y * y) / (2 * A * B)
  qp(1) = (Sqr(-A ^ 2 * x ^ 2 * (A ^ 2 * A ^ 2 + (-B ^ 2 + x ^ 2 + y ^ 2) ^ 2 - 2 * A ^ 2 * (B ^ 2 + x ^ 2 +
y ^ 2)))) / (2 * A ^ 2 * B * x)
  Call rota(pp(), u1())           ' ejecuta una rotación
  es1ap(0) = A * Rpo(0)         'Coordenadas del codo
  es1ap(1) = A * Rpo(1)

```

[La siguiente estructura condicional, permite determinar el valor del ángulo asociado a cada eslabón]

```

If pp(1) > 0 And pp(0) > 0 Then
  theta1u = Atn(pp(1) / pp(0)) * 180 / 3.1416
  Elseif pp(1) > 0 And pp(0) < 0 Then
    theta1u = (3.1416 + Atn(pp(1) / pp(0))) * 180 / 3.1416
  Elseif pp(1) < 0 And pp(0) < 0 Then
    theta1u = (3.1416 + Atn(pp(1) / pp(0))) * 180 / 3.1416

```

```

Elseif pp(1) < 0 And pp(0) > 0 Then
  theta1u = ((2 * 3.1416) + Atn(pp(1) / pp(0))) * 180 / 3.1416
Elseif pp(0) = 0 Then
  theta1u = (90)
End If

If qp(1) > 0 And qp(0) > 0 Then
  theta2u = Atn(qp(1) / qp(0)) * 180 / 3.1416
Elseif qp(1) > 0 And qp(0) < 0 Then
  theta2u = (3.1416 + Atn(qp(1) / qp(0))) * 180 / 3.1416
Elseif qp(1) < 0 And qp(0) < 0 Then
  theta2u = (3.1416 + Atn(qp(1) / qp(0))) * 180 / 3.1416
Elseif qp(1) < 0 And qp(0) > 0 Then
  theta2u = ((2 * 3.1416) + Atn(qp(1) / qp(0))) * 180 / 3.1416
Elseif qp(0) = 0 Then
  theta2u = (90)
End If

theta1_f = theta1u           ' ángulo del eslabón L1
theta2_f = theta2u         ' ángulo del eslabón L1

Call rota(qp(), Rpo())
pnd1p(0) = es1ap(0) + B * Rpo(0)           'Coordenadas del codo
pnd1p(1) = es1ap(1) + B * Rpo(1)

If Vcontrol = 1 Then
  Call PintarCD
End If

```

```

End If
End Function

```

[Esta función mantiene en pantalla, la última posición del robot]

```

Sub FijarPosición()
  Picture1.Line (0, 0)-(es1a(0), es1a(1)), QBColor(9)
  Picture1.Line (es1a(0), es1a(1))-(pnd1(0), pnd1(1)), QBColor(9)
  Picture1.Circle (es1a(0), es1a(1)), 0.3, QBColor(9)
End Sub

```

[Esta función es el modelo de la cinemática directa]

```

Sub posición(n, u, v, w)
  R0 = n * u1(0) - u * u1(1)
  R1 = n * u1(1) + u * u1(0)
  Rp0 = R0
  Rp1 = R1
  es1a(0) = A * Rp0
  es1a(1) = A * Rp1
  Ro0 = v * Rp0 - w * Rp1
  Ro1 = v * Rp1 + w * Rp0

```

```

Rp0 = Ro0
Rp1 = Ro1
pnd1(0) = es1a(0) + B * Rp0
pnd1(1) = es1a(1) + B * Rp1

```

End Sub

[la continuación la función denominada rota, aplica la operación de rotación, mientras que la función reflejar aplica la operación de reflexión, necesarias para la cinemática inversa]

Function rota(r() As Single, s() As Single)

```

Ro(0) = r(0) * s(0) - r(1) * s(1)
Ro(1) = r(0) * s(1) + r(1) * s(0)
Rpo(0) = Ro(0)
Rpo(1) = Ro(1)

```

End Function

Function reflejar(r() As Single, s() As Single)

```

Ro(0) = -r(0) * s(0) + r(1) * s(1)
Ro(1) = r(0) * s(1) + r(1) * s(0)
Rpo(0) = Ro(0)
Rpo(1) = Ro(1)

```

End Function

[Con la función borrar, las figuras en la pantalla son pintadas de igual color al fondo, para generar la impresión de movimiento]

Sub borrar()

```

Picture1.Line (0, 0)-(es1a(0), es1a(1)), QBColor(8)
Picture1.Line (es1a(0), es1a(1))-(pnd1(0), pnd1(1)), QBColor(8)
Picture1.Circle (es1a(0), es1a(1)), 0.3, QBColor(8)
Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(8)
Picture1.Line (0, 0)-(es1ap(0), es1ap(1)), QBColor(8)
Picture1.Line (es1ap(0), es1ap(1))-(pnd1p(0), pnd1p(1)), QBColor(8)
Picture1.Circle (es1ap(0), es1ap(1)), 0.3, QBColor(8)
Picture1.Circle (pnd1p(0), pnd1p(1)), 0.1, QBColor(8)
Picture1.Circle (pnd1p(0), pnd1p(1)), 0.2, QBColor(1)

```

End Sub

En este capítulo se han programado las diferentes expresiones que se establecieron para caracterizar: la cinemática directa e inversa del robot R_0 , las diferentes secuencias que pueden ser realizadas por sus actuadores para pasar de un punto dado a otro definiendo el lugar geométrico trazado por su órgano efector, las nuevas rutas de trabajo para lograr la evasión de obstáculos, así como las diferentes configuraciones de codo.

El área de trabajo, el obstáculo, la ruta de trabajo, así como el robot, se presentan en una interfase gráfica utilizando un modelo de alambre para la representación del robot y simulando el comportamiento del sistema cuando el órgano efector del multicuerpo sigue la ruta de trabajo original o la ruta modificada, de acuerdo con la presencia de un obstáculo de posición arbitraria, todo lo anteriormente dicho, puede ser observado al ejecutar el programa Ro2GDL, generado en esta tesis.

En la siguiente sección, se establecerán las conclusiones generadas sobre los diferentes tópicos tratados en el presente trabajo de tesis; es decir: caracterización de la cinemática de posición de un robot de 2 GDL, secuencias de movimientos, y el problema de evasión de obstáculos.

Conclusiones

En el presente trabajo de tesis, fueron satisfechos los siguientes objetivos:

- 1) Se establecieron modelos matemáticos representativos de un robot de 2 GDL con juntas rotacionales, aplicando las operaciones de rotación y reflexión.
- 2) Se elaboró un módulo de programación donde se simulan los diferentes tipos de movimientos que puede realizar el robot de 2 GDL, permitiendo observar el lugar geométrico descrito por el órgano efector al pasar de la posición no deformada a la deformada.
- 3) Se determinaron cinco criterios para establecer nuevas trayectorias de trabajo para el robot, logrando evadir un obstáculo presente en el área de trabajo.
- 4) Utilizando los 5 criterios de evasión, se establecieron algoritmos para la determinación de nuevas trayectorias de trabajo libres de obstáculos.
- 5) Se elaboró un módulo de programación donde se simula el movimiento del robot cubriendo una trayectoria de trabajo y mostrando su capacidad de evasión de obstáculos.

El marco teórico generado permite definir una metodología para establecer parámetros que indican la forma de obtener trayectorias libres de obstáculos; esto es, primeramente se restringe el área real de trabajo del robot a un área que realmente cubrirá de acuerdo a la trayectoria de trabajo asignada. Seguidamente se consideran las posiciones límites entre el objeto y el robot con el órgano efector en cada uno de los puntos de la trayectoria, este procedimiento permite, discriminar las posibles áreas de localización del objeto y el robot:

- a) Un área libre de obstáculos por lo que en ella no existe riesgo de colisión.
- b) Un área de posición límite entre el robot y el obstáculo lo cual implica la existencia de un solo punto de contacto.

- c) Un área de colisión donde el obstáculo debe ser evadido.

Por otro lado los programas de simulación desarrollados, permiten estudiar con detalle el comportamiento del robot al ejecutar diferentes secuencias de movimientos, así mismo como identificar las configuraciones de codo con las cuales mantener una continuidad en la trayectoria seguida por el órgano efector, resultando muy útiles para seguir un cierto contorno. El conocimiento de estas características del robot a través de la simulación, permite seleccionar las secuencias y configuraciones mas convenientes para la ejecución de las tareas que pueden ser implementadas por un sistema real.

Para futuros trabajos relacionados con este trabajo de tesis, pueden ser considerados los siguientes:

- i) Implementar los algoritmos de evasión en un sistema con control en tiempo real, para ello solo se requiere el diseño de un medio de sensado que proporcione información sobre la posición del obstáculo.
- ii) Determinar algoritmos con un alto grado de eficiencia que permitan el control en tiempo real.
- iii) Generalizar los algoritmos para objetos de diferente geometría.
- iv) Generalizar la metodología para el estudio del problema de evasión de obstáculos en tres dimensiones.

REFERENCIAS

- [1] Mark Spong, (1998) Robot Dynamics and control.
- [2] Grupta Kamal, Jung Derek (1997) octree-based Hierarchical distance maps for collision detection. Journal of Robotics Systems.
- [3] Kuu-Young, Chun-Yuh Huang, (1995) Collision-free Path Planning and Modification Based on Task Requirements .
- [4] Z. H. Zhu, R. V. Mayorga, A. K. C. Wong, (1998) Dynamic Robot Manipulator Trajectory Planning for Obstacle Avoidance.
- [5] S. W. Lee, (1997) Sub-Optimal Collision-Free Motion Planning of a two-robot system Using free Arc .
- [6] Jiménez E. Simulación de un proceso de manufactura con obstáculo en la línea de producción. México. 1998. (Tesis de Maestría en Ingeniería Mecánica). UNAM, Facultad de Ingeniería, División de Estudios de Posgrado, Sección Mecánica.
- [7] Jiménez, E, Reyes, L., Flores O., Ochoa F. (2003). Aplicaciones de las rotaciones y reflexiones a un problema de evasión de obstáculos. Informe interno de investigación DEPMI, UNAM. ISBN (En edición).
- [8] A. Rubio. Evasión de Obstáculos Mediante Multicuerpos Rígidos. México. (2000). Tesis (Maestría en Ingeniería Mecánica). UNAM, Facultad de Ingeniería, División de Estudios de Posgrado, Sección Mecánica.
- [9] Reyes Ávila Luis. Quaternions: Une Representation Parametrique Systematique Des Rotations Finies. Partie 1: Le Cadre Theorique. (1990). Rapport de Recherche INRIA Rocquencourt. France.

[10] Jiménez E., Cuenca F. Reyes L. Análisis de las secuencias de movimiento de un robot de dos grados de libertad. Coloquio 1999, Facultad de Ingeniería, UNAM (1999).

[11] Reyes, L .(1998). Sobre la parametrización de las Rotaciones y Reflexiones de Multicuerpos Rígidos en el plano. Estudios Ocasionales. Universidad Anáhuac del Sur. ISBN 968-64-02-08-X.

[12] Flores O. Modelación y prueba de secuencias de movimientos de un robot de dos grados de libertad para un problema de evasión de obstáculos en le plano. México. (2003). Tesis (Maestría en Ingeniería Mecánica). UNAM, Facultad de Ingeniería, División de Estudios de Posgrado, Sección Mecánica.

[13] Pérez A. Caracterización y prueba de las secuencias simultáneas de un robot paralelo experimental tipo RRR. México. (2001) (Tesis de Maestría en Ingeniería Mecánica). UNAM, Facultad de Ingeniería, División de Estudios de Posgrado, Sección Mecánica.

[14] Stephen Wolfram (1996) The Matemática Book, Cambridge University Press ISBN 0-521-64314-7

[15] Cohen M., Nagel E. *Introducción a la lógica y al método científico: Lógica aplicada y método científico II.* (1961). Harcour, Brace World Inc., Argentina.

[16] Cohen M., Nagel E. *Introducción a la lógica y al método científico: Lógica aplicada y método científico I.* (1961). Harcour, Brace World Inc., Argentina.

[17] Jiménez, E, Reyes, L et al. (2002). Sobre la parametrización de las rotaciones y reflexiones de multicuerpos rígidos en el plano: Modelación cinemática de un robot de dos grados de libertad. Informe interno de investigación. ISBN 968 - 36 - 9841 – 7.

[18] Roy Festherson, (1997) Robot dynamics algorithms.

[19] Yoram Koren (1990) Robotics for engineers.

[20] Mark Steven (2000) La esencia de Visual Basic.

[21] Fu K. S., González R. C., Lee C. S. G. (1987) Robotics: Control, Sensing, Vision and Intelligence.

[22] Geoffrey Gordon (1999) Simulación de sistemas.

[23] Microsoft Corporation (1999) USA, Visual Basic 6.0, manual de programación.

APÉNDICE A

ÁLGEBRA DE COMPLEJOS

Sobre la parametrización de las rotaciones y reflexiones rígidas en el plano.

1.1 El espacio vectorial de los números complejos.

En esta parte se definirán dos operaciones binarias sobre el conjunto \mathbb{R}^2 , el cual como es bien sabido, es el conjunto de parejas ordenadas de números reales, una es la operación aditiva $\oplus : \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}^2$ usual en \mathbb{R}^2 y la otra una operación multiplicativa $*$: $\mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}^2$. Se demostrará que mediante estas operaciones la pareja (\mathbb{R}^2, \oplus) y la terna $(\mathbb{R}^2, \oplus, *)$ forman un grupo aditivo conmutativo y un campo conmutativo, respectivamente. Al utilizar la multiplicación escalar $\bullet : \mathbb{R} \times \mathbb{R}^2 \rightarrow \mathbb{R}^2$ y el producto interno $\langle \bullet, \bullet \rangle : \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$ usuales, se mostrará que el conjunto $(\mathbb{R}^2, \oplus, *, \bullet, \langle \bullet, \bullet \rangle)$ es un espacio vectorial con producto interno conocido como el conjunto de números complejos (Reyes L., 1998). Considérese entonces el conjunto \mathbb{R}^2 sobre el cual se definirán las operaciones siguientes:

$$\begin{aligned} \text{i) } & (a, b) \oplus (\alpha, \beta) = (a + \alpha, b + \beta) \\ \text{ii) } & (a, b) * (\alpha, \beta) = (a\alpha - b\beta, a\beta + b\alpha), \end{aligned} \tag{1.1}$$

$$\forall (a, b), (\alpha, \beta) \in \mathbb{R}^2,$$

de hecho la operación $\oplus : \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}^2$ es la suma usual en \mathbb{R}^2 y es bien conocido que la pareja (\mathbb{R}^2, \oplus) es un grupo aditivo conmutativo. Se demuestra ahora un resultado fundamental para nuestros objetivos.

Teorema 1.1 La terna $(\mathbb{R}^2, \oplus, *)$ es un campo conmutativo.

Demostración. Puesto que la pareja (\mathbb{R}^2, \oplus) es un grupo aditivo conmutativo, falta únicamente mostrar que $(\mathbb{R}^2, *)$ es un grupo multiplicativo conmutativo excepto por la existencia del inverso multiplicativo del elemento nulo aditivo, y que además la operación aditiva y la multiplicativa satisfacen las correspondientes propiedades

distributivas. En efecto, de acuerdo a la conmutatividad de la multiplicación de los números reales, la operación $*$: $\mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}^2$ es conmutativa además:

i) la operación $*$: $\mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}^2$ es asociativa. En efecto, sean $\mathbf{p}, \mathbf{q}, \mathbf{r} \in \mathbb{R}^2$ tal que

$\mathbf{p} = (a,b), \mathbf{q} = (\alpha,\beta), \mathbf{r} = (c,d)$, entonces:

$$\begin{aligned}
 \mathbf{p} * (\mathbf{q} * \mathbf{r}) &= (a, b) * ((\alpha, \beta) * (c, d)) \\
 &= (a, b) * (\alpha c - \beta d, \alpha d + \beta c) \\
 &= (a(\alpha c - \beta d) - b(\alpha d + \beta c), a(\alpha d + \beta c) + b(\alpha c - \beta d)) \\
 &= (a\alpha c - a\beta d - b\alpha d - b\beta c, a\alpha d + a\beta c + b\alpha c - b\beta d) \\
 &= ((\alpha\alpha - b\beta)c - (a\beta + b\alpha)d, (a\beta + b\alpha)c + (\alpha\alpha - b\beta)d) \\
 &= (\alpha\alpha - b\beta, a\beta + b\alpha) * (c, d) \\
 &= ((a, b) * (\alpha, \beta)) * (c, d) \\
 &= (\mathbf{p} * \mathbf{q}) * \mathbf{r}
 \end{aligned}$$

ii) Ahora se construirá el elemento identidad de la operación multiplicación.

Se sabe que $\tilde{\mathbf{1}} \in \mathbb{R}^2$ es la identidad multiplicativa de \mathbb{R}^2 si y solo si

$\tilde{\mathbf{1}} * \mathbf{p} = \mathbf{p} * \tilde{\mathbf{1}} = \mathbf{p}, \forall \mathbf{p} \in \mathbb{R}^2$. Por tanto, siendo $\mathbf{p} = (a, b)$ y $\tilde{\mathbf{1}} = (\alpha, \beta)$, entonces,

$\tilde{\mathbf{1}} * \mathbf{p} = \mathbf{p} * \tilde{\mathbf{1}} = (a\alpha - b\beta, a\beta + b\alpha) = (a,b)$. Esto es,

$$a\alpha - b\beta = a, \quad a\beta + b\alpha = b.$$

El elemento identidad multiplicativo de $\mathbf{p} \in \mathbb{R}^2$ con respecto a la operación $*$: $\mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}^2$, se obtiene al resolver este último sistema de ecuaciones. Es decir, siendo $a \neq 0$, las dos ecuaciones se reducen a :

$$\alpha = \frac{1}{a} \{ a + b\beta \} \quad , \quad a\beta + \frac{b}{a} \{ a + b\beta \} = b.$$

Por tanto, de la segunda de estas últimas ecuaciones se obtiene que:

$$(a^2 + b^2) \beta = 0.$$

Por tanto, $\beta = 0$ y $\alpha = 1$. Esto es $\underline{\mathbf{1}} = (1, 0)$.

iii) se mostrará ahora que para toda $\mathbf{p} \in \mathfrak{R}^2$ existe $\mathbf{p}' \in \mathfrak{R}^2$, tal que $\mathbf{p} * \mathbf{p}' = \mathbf{p}' * \mathbf{p} = \underline{\mathbf{1}}$.

En efecto, siendo $\mathbf{p} = (a, b)$ y $\mathbf{p}' = (\alpha, \beta)$, entonces,

$$\mathbf{p} * \mathbf{p}' = \mathbf{p}' * \mathbf{p} = (a\alpha - b\beta, a\beta + b\alpha) = (1, 0).$$

Por tanto,

$$a\alpha - b\beta = 1 \quad , \quad a\beta + b\alpha = 0.$$

Resolviendo este sistema de ecuaciones se obtiene primeramente que,

$$\alpha = \frac{1}{a} (1 + b\beta) \quad , \quad a\beta + \frac{b}{a} (1 + b\beta) = 0.$$

Esto es, $(a^2 + b^2) \beta = -b$, equivalentemente $\beta = -\frac{b}{a^2 + b^2}$. Además, $\alpha = \frac{a}{a^2 + b^2}$.

Por tanto, el elemento inverso multiplicativo de $\mathbf{p} \in \mathfrak{R}^2$, está dado por:

$$\mathbf{p}' = \left(\frac{a}{a^2 + b^2}, \frac{b}{a^2 + b^2} \right).$$

iv) Sean ahora $\mathbf{p} = (a,b) \in \mathfrak{R}^2$, $\mathbf{q} = (\alpha, \beta) \in \mathfrak{R}^2$, y $\mathbf{r} = (c,d) \in \mathfrak{R}^2$, entonces las siguientes propiedades distributivas son satisfechas:

$$\mathbf{p} * (\mathbf{q} \oplus \mathbf{r}) = \mathbf{p} * \mathbf{q} \oplus \mathbf{p} * \mathbf{r},$$

$$(\mathbf{p} \oplus \mathbf{q}) * \mathbf{r} = \mathbf{p} * \mathbf{r} \oplus \mathbf{q} * \mathbf{r}.$$

En efecto,

$$\begin{aligned} \mathbf{p} * (\mathbf{q} \oplus \mathbf{r}) &= (a,b) * \{(\alpha, \beta) \oplus (c,d)\} = (a,b) * (\alpha + c, \beta + d) \\ &= (a(\alpha+c) - b(\beta + d), a(\beta + d) + b(\alpha+c)) \\ &= (a\alpha + ac - b\beta - bd, a\beta + ad + b\alpha + bc) \\ &= (a\alpha - b\beta, a\beta + b\alpha) \oplus (ac - bd, ad + bc) \\ &= (a,b) * (\alpha, \beta) \oplus (a,b) * (c,d) = \mathbf{p} * \mathbf{q} \oplus \mathbf{p} * \mathbf{r}. \end{aligned}$$

Además,

$$\begin{aligned} (\mathbf{p} \oplus \mathbf{q}) * \mathbf{r} &= ((a,b) \oplus (\alpha, \beta)) * (c,d) \\ &= (a + \alpha, b + \beta) * (c,d) = ((a + \alpha)c - (b + \beta)d, (a + \alpha)d + (b + \beta)c) \\ &= (ac + \alpha c - bd - \beta d, ad + \alpha d + bc + \beta c) \\ &= (ac - bd, ad + bc) \oplus (\alpha c - \beta d, \alpha d + \beta c) \end{aligned}$$

$$= (a,b) * (c,d) \oplus (\alpha, \beta) * (c,d)$$

$$= \mathbf{p} * \mathbf{r} \oplus \mathbf{q} * \mathbf{r}.$$



Es igualmente bien conocido que la operación $\bullet: \mathfrak{R} \times \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$, definida por:

$$\alpha \bullet (a,b) = (\alpha a, \alpha b), \quad \forall \alpha \in \mathfrak{R}, (a,b) \in \mathfrak{R}^2,$$

es una multiplicación escalar y por consecuencia \mathfrak{R}^2 es un espacio vectorial real. Además con esta multiplicación escalar el inverso multiplicativo de $\mathbf{p} = (a,b) \in \mathfrak{R}^2$ puede escribirse en la forma siguiente:

$$\mathbf{p}' = \frac{1}{a^2 + b^2} \bar{\mathbf{p}},$$

donde el elemento $\bar{\mathbf{p}} = (a, -b) \in \mathfrak{R}^2$ es llamado el conjugado de $\mathbf{p} = (a,b) \in \mathfrak{R}^2$. Algunas propiedades de este elemento se presentan en el teorema siguiente:

Teorema 1.2. Sea $\mathbf{p} = (a,b) \in \mathfrak{R}^2$ y $\mathbf{q} = (\alpha,b) \in \mathfrak{R}^2$, entonces:

- i) $\overline{\mathbf{p} \oplus \mathbf{q}} = \bar{\mathbf{p}} \oplus \bar{\mathbf{q}},$
- ii) $\overline{\mathbf{p} * \mathbf{q}} = \bar{\mathbf{q}} * \bar{\mathbf{p}},$
- iii) $\mathbf{p} * \bar{\mathbf{p}} = \bar{\mathbf{p}} * \mathbf{p} = (a^2 + b^2, 0).$

Demostración. Observemos primeramente que de acuerdo con la definición de la operación aditiva en \mathfrak{R}^2 y la correspondiente al conjugado se obtiene que:

$$\overline{\mathbf{p} \oplus \mathbf{q}} = \overline{(a + \alpha, b + \beta)} = (a + \alpha, -b - \beta) = (a, -b) \oplus (\alpha, -\beta) = \bar{\mathbf{p}} \oplus \bar{\mathbf{q}},$$

también,

$$\begin{aligned} \bar{\mathbf{q}} * \bar{\mathbf{p}} &= \overline{(a,b) * (\alpha,\beta)} = \overline{(a\alpha - b\beta, a\beta + b\alpha)} = (a\alpha - b\beta, -a\beta - b\alpha) = \\ &= (\alpha, -\beta) * (a,-b) = \bar{\mathbf{q}} * \bar{\mathbf{p}}. \end{aligned}$$

Finalmente,

$$\mathbf{p} * \bar{\mathbf{p}} = (a,b) * (a, -b) = (a^2 + b^2, 0).$$



Es también bien conocido, que la transformación $\langle \bullet, \bullet \rangle: \mathfrak{R}^2 \times \mathfrak{R}^2 \rightarrow \mathfrak{R}$ definida, siendo $\mathbf{p} = (a, b) \in \mathfrak{R}^2$ y $\mathbf{q} = (\alpha, \beta) \in \mathfrak{R}^2$, por:

$$\langle \mathbf{p}, \mathbf{q} \rangle = a\alpha + b\beta,$$

es un producto interno en \mathfrak{R}^2 , de tal suerte que el conjunto $\mathcal{C} = (\mathfrak{R}^2, \oplus, *, \bullet, \langle \bullet, \bullet \rangle)$ es un espacio vectorial con producto interno y la función $\|\bullet\|: \mathfrak{R}^2 \rightarrow \mathfrak{R}$ definida por:

$$\|\mathbf{p}\| = \langle \mathbf{p}, \mathbf{p} \rangle^{1/2}, \forall \mathbf{p} \in \mathfrak{R}^2,$$

es una norma, por tanto \mathcal{C} es un espacio vectorial normado, y sus elementos son llamados comúnmente números complejos.

Observación 1.1. Observemos que al tomar en cuenta la definición de la operación aditiva, el elemento $\mathbf{p} = (a, b) \in \mathcal{C}$ puede expresarse como la suma siguiente:

$$\mathbf{p} = (a, b) = (a, 0) \oplus (0, b).$$

Es decir, al definir los subespacios

$$\mathcal{C}_R = \{(a, 0): a \in \mathfrak{R}\} \subset \mathfrak{R}^2, \mathcal{C}_V = \{(0, b): b \in \mathfrak{R}\} \subset \mathfrak{R}^2,$$

del espacio vectorial de números complejos, éste puede expresarse mediante la siguiente suma directa de subespacios:

$$\mathcal{C} = \mathcal{C}_R \dot{\cup} \mathcal{C}_V \equiv \{\mathbf{p} \in \mathcal{C} : \mathbf{p} = \mathbf{p}_R \oplus \mathbf{p}_V; \mathbf{p}_R \in \mathcal{C}_R, \mathbf{p}_V \in \mathcal{C}_V\}.$$

También al definir los isomorfismos $T_R: \mathcal{C}_R \rightarrow \mathfrak{R}; T_V: \mathcal{C}_V \rightarrow \mathfrak{R}$ por:

$$T_R(a, 0) = a \quad , \quad T_V(0, b) = b,$$

y, siendo $\mathbf{p} = (a, b) \in \mathcal{C}$, entonces,

$$\mathbf{p} = T_R^{-1}(a) \oplus T_V^{-1}(b).$$

Al observar finalmente que, siendo $\mathbf{q} = (\alpha, \beta) \in \mathfrak{R}^2$,

$$\bar{\mathbf{p}} * \mathbf{q} \oplus \bar{\mathbf{q}} * \mathbf{p} = 2(a\alpha + b\beta, 0),$$

el producto interno de los elementos $\mathbf{p}, \mathbf{q} \in \mathfrak{R}^2$ puede expresarse en la siguiente forma:

$$\langle \mathbf{p}, \mathbf{q} \rangle \equiv \frac{1}{2} T_R \{ \bar{\mathbf{p}} * \mathbf{q} \oplus \bar{\mathbf{q}} * \mathbf{p} \}.$$



Observación 1.2. Obsérvese que el elemento $\underline{e}_2 = (0, 1) \in \mathfrak{R}^2$ de la base canónica, satisface que:

$$i^2 = \underline{e}_2 * \underline{e}_2 = (0, 1) * (0, 1) = (-1, 0) \equiv T_R^{-1}(-1).$$

Además, siendo $\mathbf{p} = (a, b)$, entonces,

$$\mathbf{p} = (a, b) = (a, 0) \oplus b \underline{e}_2 = T_R^{-1}(a) \oplus b \underline{e}_2.$$

Estas dos últimas expresiones han sido utilizadas para interpretar los números complejos como la suma de una parte real, mas una parte imaginaria. En realidad el número imaginario i , comúnmente utilizado en el álgebra compleja, es el vector \underline{e}_2 de la base canónica y la representación algebraicamente correcta de la pareja $\mathbf{p} = (a, b) \in \mathfrak{R}^2$, está dada por la última ecuación de esta observación.



1.2 Representación paramétrica de rotaciones.

El objetivo de esta parte es construir con la operación $*$: $\mathfrak{R}^2 \times \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$, estudiada anteriormente, una transformación lineal $\rho(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$, con $\mathbf{p} \in \mathfrak{R}^2$ fijo, que preserve el producto interno y cuya matriz con respecto a la base canónica posea determinante positivo, con lo cual dicha transformación de parámetro $\mathbf{p} \in \mathfrak{R}^2$, es una rotación. Se estudiarán también algunas otras representaciones de la rotación antes mencionada. Sea $\rho(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$, $\mathbf{p} \in \mathfrak{R}^2$, fijo definida por:

$$(a, b) * (\alpha, \beta) = (a\alpha - b\beta, a\beta + b\alpha), \quad \forall (a, b), (\alpha, \beta) \in \mathfrak{R}^2,$$

$$\rho(\mathbf{p}, \mathbf{q}) = \frac{1}{\|\mathbf{p}\|} \mathbf{p} * \mathbf{q}, \quad \forall \mathbf{q} \in \mathfrak{R}^2. \quad (1.2)$$

Las propiedades de la transformación $\rho(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$, definida anteriormente se resumen en los resultados siguientes.

Teorema 1.3. La transformación $\rho(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ definida en la ecuación [1.2] es lineal y ortogonal de determinante positivo.

Demostración: En efecto, siendo $\mathbf{q}, \mathbf{s} \in \mathfrak{R}^2$, y de acuerdo a la distributividad de la operación $*$: $\mathfrak{R}^2 \times \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$, se tiene que:

$$\begin{aligned} \rho(\mathbf{p}, \mathbf{q} \oplus \mathbf{s}) &= \frac{1}{\|\mathbf{p}\|} \mathbf{p} * (\mathbf{q} \oplus \mathbf{s}) \\ &= \frac{1}{\|\mathbf{p}\|} \{\mathbf{p} * \mathbf{q} \oplus \mathbf{p} * \mathbf{s}\} = \frac{1}{\|\mathbf{p}\|} \{\mathbf{p} * \mathbf{q}\} \oplus \frac{1}{\|\mathbf{p}\|} \{\mathbf{p} * \mathbf{s}\} = \rho(\mathbf{p}, \mathbf{q}) \oplus \rho(\mathbf{p}, \mathbf{s}), \end{aligned}$$

también, $\forall \alpha \in \mathfrak{R}$,

$$\rho(\mathbf{p}, \alpha \mathbf{q}) = \frac{1}{\|\mathbf{p}\|} \{\mathbf{p} * (\alpha \bullet \mathbf{q})\}$$

$$= \frac{\alpha}{\|\mathbf{p}\|} \bullet \{\mathbf{p} * \mathbf{q}\} = \alpha \bullet \rho(\mathbf{p}, \mathbf{q}).$$

Además, $\rho(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ es ortogonal, pues:

$$\begin{aligned} \langle \rho(\mathbf{p}, \mathbf{q}), \rho(\mathbf{p}, \mathbf{r}) \rangle &= \frac{1}{2} \text{Tr} \{ \rho(\overline{\mathbf{p}}, \mathbf{q}) * \rho(\mathbf{p}, \mathbf{r}) \oplus \rho(\overline{\mathbf{p} * \mathbf{r}}) * \rho(\mathbf{p}, \mathbf{q}) \} \\ &= \frac{1}{2} \frac{1}{\|\mathbf{p}\|^2} \text{Tr} \{ (\overline{\mathbf{p} * \mathbf{q}}) * (\mathbf{p} * \mathbf{r}) \oplus (\overline{\mathbf{p} * \mathbf{r}}) * (\mathbf{p} * \mathbf{q}) \} \\ &= \frac{1}{2} \frac{1}{\|\mathbf{p}\|^2} \text{Tr} \{ \bar{\mathbf{q}} * (\bar{\mathbf{p}} * \mathbf{p}) * \mathbf{r} \oplus \bar{\mathbf{r}} * (\bar{\mathbf{p}} * \mathbf{p}) * \mathbf{q} \} \\ &= \frac{1}{2} \text{Tr} \{ \bar{\mathbf{q}} * \mathbf{r} \oplus \bar{\mathbf{r}} * \mathbf{q} \} = \langle \mathbf{q}, \mathbf{r} \rangle. \end{aligned}$$

Finalmente se muestra que el determinante de la transformación $\rho(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ es positivo, en efecto, siendo $B = \{(1,0), (0,1)\} \subset \mathfrak{R}^2$ la base canónica en \mathfrak{R}^2 , la matriz de $\rho(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ con respecto a esta base resulta ser,

$$M_{\rho(\mathbf{p}, \bullet)} \equiv \frac{1}{\|\mathbf{p}\|} \begin{pmatrix} a & -b \\ b & a \end{pmatrix},$$

pues,

$$\begin{aligned} \rho(\mathbf{p}, \underline{e}_1) &= \frac{1}{\|\mathbf{p}\|} (a, b) * (1, 0) \equiv \frac{1}{\|\mathbf{p}\|} (a, b), \\ \rho(\mathbf{p}, \underline{e}_2) &= \frac{1}{\|\mathbf{p}\|} (a, b) * (0, 1) \equiv \frac{1}{\|\mathbf{p}\|} (-b, a). \end{aligned}$$

El resultado buscado se obtiene al observar que $\det M_{\rho(\mathbf{p}, \bullet)} = 1$.



Teorema 1.4. La transformación $\rho(\mathbf{p}, \bullet) \in L(\mathfrak{R}^2, \mathfrak{R}^2) = \{\underline{T}: \mathfrak{R}^2 \rightarrow \mathfrak{R}^2: \underline{T} \text{ es lineal}\}$, puede ser representada en la forma siguiente:

$$\rho(\mathbf{p}, \bullet) = \frac{1}{\|\mathbf{p}\|} \{a \underline{\mathbf{I}} + b \underline{\mathbf{W}}\} \equiv \frac{1}{\|\mathbf{p}\|} \{b \underline{\mathbf{W}} - a \underline{\mathbf{W}}^2\},$$

donde $\underline{\mathbf{I}}: \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ es la transformación identidad y $\underline{\mathbf{W}}: \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ está dado por:

$$\underline{\mathbf{W}} = -\underline{\mathbf{e}}_1 \otimes \underline{\mathbf{e}}_2 + \underline{\mathbf{e}}_2 \otimes \underline{\mathbf{e}}_1.$$

Demostración. En efecto, observemos que:

$$M_{\rho(\mathbf{p}, \bullet)} \equiv \frac{1}{\|\mathbf{p}\|} \left\{ \begin{pmatrix} a & 0 \\ 0 & a \end{pmatrix} + \begin{pmatrix} 0 & -b \\ b & 0 \end{pmatrix} \right\} \equiv \frac{1}{\|\mathbf{p}\|} \{a \mathbf{I} + b \mathbf{W}\}$$

donde \mathbf{I} es la matriz identidad y la matriz antisimétrica \mathbf{W} está dada por:

$$\mathbf{W} \equiv \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}.$$

La segunda igualdad del teorema se obtiene al observar que:

$$\mathbf{W}^2 \equiv \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \equiv \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \equiv -\mathbf{I}.$$

■

La parte antisimétrica $\underline{\mathbf{W}} \in A(\mathfrak{R}^2, \mathfrak{R}^2) = \{\underline{\mathbf{B}}: \mathfrak{R}^2 \rightarrow \mathfrak{R}^2: \underline{\mathbf{B}} = -\underline{\mathbf{B}}^T\}$ del teorema anterior, satisface el siguiente resultado:

Teorema 1.5. La transformación $\tilde{W} \in A(\mathfrak{R}^2, \mathfrak{R}^2)$ es tal que

$$\tilde{W}^{2n-1} = (-1)^{n-1} \tilde{W} \quad , \quad \tilde{W}^{2n} = (-1)^{n-1} \tilde{W}^2 \quad , \quad n = 2, 3, 4, \dots$$

Demostración. Para demostrar este teorema se procede por inducción, mostrando primeramente que para $n = 2$ el resultado es satisfecho. En efecto,

$$\tilde{W}^3 = \tilde{W}^2 \circ \tilde{W} = -\tilde{I} \circ \tilde{W} \equiv -\tilde{W} \quad , \quad \tilde{W}^4 = \tilde{W}^3 \circ \tilde{W} = -\tilde{W} \circ \tilde{W} = -\tilde{W}^2 \quad ,$$

por tanto para $n = 2$, el resultado es obtenido. Supongamos ahora que para $n-1$ el resultado se satisface, entonces

$$\begin{aligned} \tilde{W}^{2n-1} &= \tilde{W}^{2n-2} \circ \tilde{W} = \tilde{W}^{2(n-1)} \circ \tilde{W} = \tilde{W}^{2(n-1)-1} \circ \tilde{W}^2 \\ &= (-1)^{n-2} \tilde{W} \circ \tilde{W}^2 = (-1)^{n-2} \tilde{W}^3 = -(-1)^{n-2} \tilde{W} = \\ &= (-1)^{n-1} \tilde{W} . \end{aligned}$$

Además,

$$\begin{aligned} \tilde{W}^{2n} &= \tilde{W}^{2n-2} \circ \tilde{W}^2 = \tilde{W}^{2(n-1)} \circ \tilde{W}^2 = (-1)^{n-2} \tilde{W}^2 \circ \tilde{W}^2 \\ &= (-1)^{n-2} \tilde{W}^4 = (-1)^{n-2} (-1) \tilde{W}^2 = (-1)^{n-1} \tilde{W}^2 . \end{aligned}$$

■

1.3 Representación exponencial de $\rho(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$.

Para interpretar físicamente los parámetros de la rotación $\rho(\mathbf{p}, \bullet) \in L(\mathfrak{R}^2, \mathfrak{R}^2)$, se utilizará la geometría del espacio vectorial de los números complejos, presentando el resultado fundamental en el siguiente teorema.

Teorema 1.6. Sea $\mathbf{u} \in \mathfrak{R}^2$, $\mathbf{p} = (a, b) \in \mathfrak{R}^2$ y $\mathbf{v} = \rho(\mathbf{p}, \mathbf{u}) \in \mathfrak{R}^2$. Entonces el coseno del ángulo formado por los vectores \mathbf{u} y \mathbf{v} está dado por:

$$\text{Cos } \theta = \frac{a}{\|\mathbf{p}\|} \quad , \quad \text{Sen } \theta = \pm \frac{b}{\|\mathbf{p}\|} .$$

Demostración. En efecto, el coseno del ángulo está definido en términos del producto interno del vector $\mathbf{u} \in \mathfrak{R}^2$ y el vector rotado $\mathbf{v} = \rho(\mathbf{p}, \mathbf{u}) \in \mathfrak{R}^2$, esto es:

$$\text{Cos } \theta = \frac{\langle \mathbf{u}, \rho(\mathbf{p}, \mathbf{u}) \rangle}{\|\mathbf{u}\| \|\rho(\mathbf{p}, \mathbf{u})\|} .$$

Además, siendo $\rho(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ una rotación, entonces $\|\mathbf{u}\| = \|\rho(\mathbf{p}, \mathbf{u})\|$. También,

$$\rho(\mathbf{p}, \mathbf{u}) = \frac{1}{\|\mathbf{p}\|} (a u_1 - b u_2, a u_2 + b u_1),$$

por tanto,

$$\langle \mathbf{u}, \rho(\mathbf{p}, \mathbf{u}) \rangle = \frac{1}{\|\mathbf{p}\|} (a u_1^2 - b u_1 u_2 + a u_2^2 + b u_1 u_2) = \frac{a \|\mathbf{u}\|^2}{\|\mathbf{p}\|} .$$

Esto es,

$$\text{Cos } \theta = \frac{a \|\mathbf{u}\|^2}{\|\mathbf{p}\| \|\mathbf{u}\|^2} = \frac{a}{\|\mathbf{p}\|} .$$

Finalmente, de la igualdad $\text{Cos}^2 \theta + \text{Sen}^2 \theta = 1$ se obtiene que:

$$\text{Sen}^2 \theta = 1 - \text{Cos}^2 \theta = 1 - \frac{a^2}{\|\mathbf{p}\|^2} \equiv \frac{1}{\|\mathbf{p}\|^2} \left\{ \|\mathbf{p}\|^2 - a^2 \right\} = \frac{b^2}{\|\mathbf{p}\|^2} ,$$

por tanto, $\text{Sen } \theta = \pm \frac{b}{\|\mathbf{p}\|}$.

■

El resultado del teorema anterior nos permite encontrar la siguiente representación trigonométrica de la rotación $\rho(\mathbf{p}, \bullet) \in L(\mathfrak{R}^2, \mathfrak{R}^2)$.

Corolario 1.1 La transformación $\rho(\mathbf{p}, \bullet) \in L(\mathfrak{R}^2, \mathfrak{R}^2)$ puede ser representada en la forma:

$$\rho(\mathbf{p}, \bullet) = \text{Cos } \theta \tilde{\mathbf{I}} \pm \text{Sen } \theta \tilde{\mathbf{W}}.$$

Demostración. En efecto, de acuerdo al teorema 1.6 tenemos que $a = \|\mathbf{p}\| \text{Cos } \theta$ y $b = \pm \|\mathbf{p}\| \text{Sen } \theta$, entonces,

$$\begin{aligned} \rho(\mathbf{p}, \bullet) &= \frac{1}{\|\mathbf{p}\|} \{ a \tilde{\mathbf{I}} + b \tilde{\mathbf{W}} \} \equiv \frac{1}{\|\mathbf{p}\|} \{ \|\mathbf{p}\| \text{Cos } \theta \tilde{\mathbf{I}} \pm \|\mathbf{p}\| \text{Sen } \theta \tilde{\mathbf{W}} \} \\ &= \text{Cos } \theta \tilde{\mathbf{I}} \pm \text{Sen } \theta \tilde{\mathbf{W}}. \end{aligned}$$

■

Corolario 1.2. La transformación $\rho(\mathbf{p}, \bullet) \in L(\mathfrak{R}^2, \mathfrak{R}^2)$ es tal que:

$$\rho(\mathbf{p}, \bullet) = \sum_{n=0}^{\infty} (\pm 1)^n \frac{1}{n!} \tilde{\mathbf{W}}_{\theta}^n.$$

Demostración. En efecto, recordemos que la transformación $\rho(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ puede expresarse de acuerdo con el teorema anterior en la forma:

$$\rho(\mathbf{p}, \bullet) = \text{Cos } \theta \tilde{\mathbf{I}} \pm \text{Sen } \theta \tilde{\mathbf{W}}.$$

Al considerar en esta expresión las expansiones en series de las funciones trigonométricas, esto es,

$$\text{Sen } \theta = \theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!} - \frac{\theta^7}{7!} + \dots, \quad \text{Cos } \theta = 1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!} - \frac{\theta^6}{6!} + \dots$$

Se obtiene, siendo $\tilde{W}_\theta = \theta \tilde{W}$, que:

Caso 1.

$$\rho(\mathbf{p}, \bullet) = \cos \theta \tilde{\mathbf{I}} + \operatorname{Sen} \theta \tilde{\mathbf{W}}$$

$$\begin{aligned} &= \left(1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!} - \frac{\theta^6}{6!} + \dots\right) \tilde{\mathbf{I}} + \left(\theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!} - \frac{\theta^7}{7!} + \dots\right) \tilde{\mathbf{W}} \\ &= \tilde{\mathbf{I}} + \theta \tilde{\mathbf{W}} - \frac{\theta^2}{2!} \tilde{\mathbf{I}} - \frac{\theta^3}{3!} \tilde{\mathbf{W}} + \frac{\theta^4}{4!} \tilde{\mathbf{I}} + \frac{\theta^5}{5!} \tilde{\mathbf{W}} - \frac{\theta^6}{6!} \tilde{\mathbf{I}} - \frac{\theta^7}{7!} \tilde{\mathbf{W}} + \dots \\ &= \tilde{\mathbf{I}} + \theta \tilde{\mathbf{W}} + \frac{\theta^2}{2!} \tilde{\mathbf{W}}^2 + \frac{\theta^3}{3!} \tilde{\mathbf{W}}^3 + \frac{\theta^4}{4!} \tilde{\mathbf{W}}^4 + \frac{\theta^5}{5!} \tilde{\mathbf{W}}^5 + \frac{\theta^6}{6!} \tilde{\mathbf{W}}^6 + \frac{\theta^7}{7!} \tilde{\mathbf{W}}^7 + \dots \\ &= \sum_{n=0}^{\infty} \frac{1}{n!} \tilde{\mathbf{W}}_\theta^n. \end{aligned}$$

Caso 2.

$$\rho(\mathbf{p}, \bullet) = \cos \theta \tilde{\mathbf{I}} - \operatorname{Sen} \theta \tilde{\mathbf{W}}$$

$$\begin{aligned} &= \left(1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!} - \frac{\theta^6}{6!} + \dots\right) \tilde{\mathbf{I}} - \left(\theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!} - \frac{\theta^7}{7!} + \dots\right) \tilde{\mathbf{W}} \\ &= \tilde{\mathbf{I}} - \theta \tilde{\mathbf{W}} - \frac{\theta^2}{2!} \tilde{\mathbf{I}} + \frac{\theta^3}{3!} \tilde{\mathbf{W}} + \frac{\theta^4}{4!} \tilde{\mathbf{I}} - \frac{\theta^5}{5!} \tilde{\mathbf{W}} - \frac{\theta^6}{6!} \tilde{\mathbf{I}} + \frac{\theta^7}{7!} \tilde{\mathbf{W}} + \dots \\ &= \tilde{\mathbf{I}} - \theta \tilde{\mathbf{W}} + \frac{\theta^2}{2!} \tilde{\mathbf{W}}^2 - \frac{\theta^3}{3!} \tilde{\mathbf{W}}^3 + \frac{\theta^4}{4!} \tilde{\mathbf{W}}^4 - \frac{\theta^5}{5!} \tilde{\mathbf{W}}^5 + \frac{\theta^6}{6!} \tilde{\mathbf{W}}^6 - \frac{\theta^7}{7!} \tilde{\mathbf{W}}^7 + \dots \\ &= \tilde{\mathbf{I}} - \frac{\tilde{\mathbf{W}}_\theta}{\theta} + \frac{1}{2!} \frac{\tilde{\mathbf{W}}_\theta^2}{\theta^2} - \frac{1}{3!} \frac{\tilde{\mathbf{W}}_\theta^3}{\theta^3} + \frac{1}{4!} \frac{\tilde{\mathbf{W}}_\theta^4}{\theta^4} - \frac{\theta^5}{5!} \frac{\tilde{\mathbf{W}}_\theta^5}{\theta^5} + \frac{1}{6!} \frac{\tilde{\mathbf{W}}_\theta^6}{\theta^6} - \frac{1}{7!} \frac{\tilde{\mathbf{W}}_\theta^7}{\theta^7} + \dots \end{aligned}$$

$$= \sum_{n=0}^{\infty} (-1)^n \frac{1}{n!} \mathbf{W}^n \cdot$$

■

Teorema 1.7. Sea $\mathbf{q} = (q_0, q_1) \in \mathbb{R}^2$, $\mathbf{p} = (a, b) \in \mathbb{R}^2$, entonces la siguiente igualdad es satisfecha:

$$\mathbf{q} \times \rho(\mathbf{p}, \mathbf{q}) \equiv \frac{b}{\|\mathbf{p}\|} \|\mathbf{q}\|^2 \mathbf{e}_3 \equiv (0, 0, \frac{b}{\|\mathbf{p}\|} \|\mathbf{q}\|^2).$$

Demostración. En efecto,

$$\begin{aligned} \mathbf{q} \times \rho(\mathbf{p}, \mathbf{q}) &\equiv \varepsilon_{ijk} q_j \rho(\mathbf{p}, \mathbf{q})_k \mathbf{e}_i \equiv \varepsilon_{3jk} q_j \rho(\mathbf{p}, \mathbf{q})_k \mathbf{e}_3 \\ &\equiv \{\varepsilon_{31k} q_0 \rho(\mathbf{p}, \mathbf{q})_k + \varepsilon_{32k} q_1 \rho(\mathbf{p}, \mathbf{q})_k\} \mathbf{e}_3 \\ &\equiv \{\varepsilon_{312} q_0 \rho(\mathbf{p}, \mathbf{q})_2 + \varepsilon_{321} q_1 \rho(\mathbf{p}, \mathbf{q})_1\} \mathbf{e}_3 \\ &\equiv \{q_0 \rho(\mathbf{p}, \mathbf{q})_2 - q_1 \rho(\mathbf{p}, \mathbf{q})_1\} \mathbf{e}_3 \\ &\equiv \frac{1}{\|\mathbf{p}\|} \{a q_0 q_1 + b q_0^2 - a q_0 q_1 - b q_1^2\} \mathbf{e}_3 \\ &\equiv \frac{b}{\|\mathbf{p}\|} \|\mathbf{q}\|^2 \mathbf{e}_3. \end{aligned}$$

■

Corolario 1.3. Sea $\mathbf{q} = (q_0, q_1) \in \mathbb{R}^2$, $\mathbf{p} = (a, b) \in \mathbb{R}^2$, entonces:

$$\mathbf{q} \times \rho(\mathbf{p}, \mathbf{q}) \equiv (0, 0, \pm \text{Sen } \theta \frac{\|\mathbf{q}\|^2}{\|\mathbf{p}\|}).$$

Demostración. En efecto el resultado buscado se obtiene de combinar los teoremas 1.6 y 1.7.



1.4 Otra operación binaria multiplicativa.

En esta parte se presentará otra operación multiplicativa $\otimes: \mathfrak{R}^2 \times \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$, mediante la cual podremos construir una transformación $\rho_2(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$, $\mathbf{p} \in \mathfrak{R}^2$ fijo, que resultará ser una reflexión. Consideremos entonces la operación

$\otimes: \mathfrak{R}^2 \times \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ definida por:

$$(a,b) \otimes (\alpha, \beta) = (-a\alpha + b\beta, a\beta + b\alpha). \quad (1.4)$$

Se estudiarán ahora las propiedades de dicha operación a través de los resultados siguientes:

Teorema 1.11. La operación $\otimes: \mathfrak{R}^2 \times \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ es no asociativa.

Demostración. En efecto, sean $\mathbf{p}=(a,b)$, $\mathbf{q}=(\alpha,\beta)$, $\mathbf{r}=(c,d)$, entonces,

$$\begin{aligned} \mathbf{p} \otimes (\mathbf{q} \otimes \mathbf{r}) &= (a,b) \otimes \{(\alpha,\beta) \otimes (c,d)\} \\ &\equiv (a,b) \otimes \{(-\alpha c + \beta d, \alpha d + \beta c)\} \\ &\equiv (-a(-\alpha c + \beta d) + b(\alpha d + \beta c), a(\alpha d + \beta c) + b(-\alpha c + \beta d)) \\ &\equiv (a\alpha c - a\beta d + b\alpha d + b\beta c, a\alpha d + a\beta c - b\alpha c + b\beta d) \\ &\equiv (c(a\alpha + b\beta) + d(b\alpha - a\beta), (a\alpha + b\beta)d + (a\beta - b\alpha)c). \end{aligned}$$

También,

$$\begin{aligned} (\mathbf{p} \otimes \mathbf{q}) \otimes \mathbf{r} &= \{(a,b) \otimes (\alpha,\beta)\} \otimes (c,d) \equiv (-a\alpha + b\beta, a\beta + b\alpha) \otimes (c,d) \\ &= (-c(-a\alpha + b\beta) + d(a\beta + b\alpha), d(-a\alpha + b\beta) + c(a\beta + b\alpha)) \end{aligned}$$

$$= (c(a\alpha - b\beta) + d(a\beta + b\alpha), d(-a\alpha + b\beta) + c(a\beta - b\alpha)).$$

Por tanto,

$$\mathbf{p} \circledast (\mathbf{q} \circledast \mathbf{r}) \neq (\mathbf{p} \circledast \mathbf{q}) \circledast \mathbf{r}.$$

■

Teorema 1.12. El elemento $\tilde{\mathbf{1}} = \frac{1}{a^2 + b^2} (b^2 - a^2, 2ba)$, es tal que si $\mathbf{p} = (a, b)$,

entonces, $\mathbf{p} \circledast \tilde{\mathbf{1}} \equiv \mathbf{p}$.

Demostración. En efecto, sean $\mathbf{p} = (a, b)$ y $\tilde{\mathbf{1}} = (\alpha, \beta)$ tal que $\mathbf{p} \circledast \tilde{\mathbf{1}} \equiv \mathbf{p}$, esto es:

$$(a, b) \circledast (\alpha, \beta) = (-a\alpha + b\beta, a\beta + b\alpha) = (a, b),$$

por tanto,

$$-a\alpha + b\beta = a, \quad a\beta + b\alpha = b.$$

Al resolver este sistema de ecuaciones, se obtiene que:

$$\alpha = -\frac{1}{a}(a - b\beta), \quad a\beta - \frac{b}{a}(a - b\beta) = b.$$

Por tanto, $\beta(a^2 + b^2) = 2ba$. Esto es $\beta = \frac{2ba}{a^2 + b^2}$. También,

$$\alpha = -\frac{1}{a}\left(a - \frac{2b^2a}{a^2 + b^2}\right) = \frac{1}{a^2 + b^2}(-a^2 - b^2 + 2b^2) \equiv \frac{b^2 - a^2}{a^2 + b^2}.$$

Por tanto, $\tilde{\mathbf{1}} = \frac{1}{a^2 + b^2} (b^2 - a^2, 2ba)$.

■

Teorema 1.13. Sea $\mathbf{p} = (a, b) \in \mathfrak{R}^2$, entonces, el elemento $\mathbf{p}' = \frac{1}{a^2 + b^2} (a, b)$, es tal que $\mathbf{p} \circledast \mathbf{p}' \equiv \mathbf{1}$.

Demostración. En efecto, sea $\mathbf{p}' \equiv (\alpha, \beta)$ tal que $\mathbf{p} * \mathbf{p}' \equiv \mathbf{1}$, esto es,

$$(a, b) \circledast (\alpha, \beta) = (-a\alpha + b\beta, a\beta + b\alpha) \equiv \frac{1}{a^2 + b^2} (b^2 - a^2, 2ba).$$

Entonces,

$$-a\alpha + b\beta \equiv \frac{1}{a^2 + b^2} (b^2 - a^2) \quad , \quad a\beta + b\alpha = 2ab.$$

Esto es,

$$\alpha = -\frac{1}{(a^2 + b^2)a} (b^2 - a^2) + \frac{b}{a}\beta.$$

Por tanto,

$$a\beta - \frac{b}{a(a^2 + b^2)} (b^2 - a^2) + \frac{b^2}{a}\beta = \frac{1}{a^2 + b^2} 2ab.$$

Esto es,

$$\left(a + \frac{b^2}{a}\right)\beta = \frac{1}{a^2 + b^2} \left(2ab + \frac{b}{a}(b^2 - a^2)\right)$$

$$= \frac{1}{a^2 + b^2} \left(ab + \frac{b^3}{a}\right) = \frac{b}{a(a^2 + b^2)} (a^2 + b^2) = \frac{b}{a}$$

por tanto,

$$\left(\frac{a^2+b^2}{a}\right)\beta = \frac{b}{a},$$

de donde, $\beta = \frac{b}{a^2+b^2}$, y

$$\alpha = -\frac{1}{a(a^2+b^2)}(b^2-a^2) + \frac{b}{a}\left(\frac{b}{a^2+b^2}\right) \equiv \frac{1}{a(a^2+b^2)}\{-b^2+a^2+b^2\}$$

$$\equiv \frac{a}{a^2+b^2}.$$

Por tanto, $\mathbf{p}' = \frac{1}{a^2+b^2}(a, b) = \frac{1}{a^2+b^2}\mathbf{p}$.

■

Teorema 1.14. Sean $\mathbf{p}, \mathbf{q}, \mathbf{r} \in \mathfrak{R}^2$. Entonces, las siguientes propiedades distributivas son satisfechas:

$$\mathbf{p} \circledast (\mathbf{q} \oplus \mathbf{r}) = \mathbf{p} \circledast \mathbf{q} \oplus \mathbf{p} \circledast \mathbf{r},$$

$$(\mathbf{p} \oplus \mathbf{q}) \circledast \mathbf{r} = \mathbf{p} \circledast \mathbf{r} \oplus \mathbf{q} \circledast \mathbf{r}.$$

Demostración. En efecto, sean $\mathbf{p} = (a, b)$, $\mathbf{q} = (\alpha, \beta)$ y $\mathbf{r} = (c, d)$, entonces

$$\begin{aligned} \mathbf{p} \circledast (\mathbf{q} \oplus \mathbf{r}) &= (a, b) \circledast (\alpha + c, \beta + d) \\ &\equiv (-a(\alpha + c) + b(\beta + d), b(\alpha + c) + a(\beta + d)) \\ &= (-a\alpha + b\beta, a\beta + b\alpha) \oplus (-ac + bd, ad + bc) \\ &= (a, b) \circledast (\alpha, \beta) \oplus (a, b) \circledast (c, d) \\ &\equiv \mathbf{p} \circledast \mathbf{q} \oplus \mathbf{p} \circledast \mathbf{r}. \end{aligned}$$

Además,

$$\begin{aligned}
(\mathbf{p} \oplus \mathbf{q}) \circledast \mathbf{r} &= (a + \alpha, b + \beta) \circledast (c, d) \\
&= (-c(a + \alpha) + d(b + \beta), d(a + \alpha) + c(b + \beta)) \\
&= (-ca + db, da + cb) \oplus (-c\alpha + d\beta, d\alpha + c\beta) \\
&= (a, b) \circledast (c, d) \oplus (\alpha, \beta) \circledast (c, d) \\
&\equiv \mathbf{p} \circledast \mathbf{r} \oplus \mathbf{q} \circledast \mathbf{r}.
\end{aligned}$$

■

Teorema 1.15. La operación $\circledast: \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}^2$ es conmutativa. En efecto si $\mathbf{p} = (a, b)$, $\mathbf{q} = (\alpha, \beta)$, entonces,

$$\begin{aligned}
\mathbf{p} \circledast \mathbf{q} &\equiv (a, b) \circledast (\alpha, \beta) \equiv (-a\alpha + b\beta, b\alpha + a\beta) = (-\alpha a + \beta b, \alpha b + \beta a) \\
&\equiv (\alpha, \beta) \circledast (a, b) \equiv \mathbf{q} \circledast \mathbf{p}.
\end{aligned}$$

■

Los siguientes resultados, son satisfechos por la operación $\circledast: \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}^2$.

Teorema 1.16. Sean $\mathbf{p} = (a, b) \in \mathbb{R}^2$, $\mathbf{q} = (\alpha, \beta) \in \mathbb{R}^2$. Entonces

- i) $\overline{\mathbf{p} \circledast \mathbf{q}} = \bar{\mathbf{q}} \circledast \bar{\mathbf{p}},$
- ii) $\mathbf{p} \circledast \bar{\mathbf{p}} = \bar{\mathbf{p}} \circledast \mathbf{p} = -(a^2 + b^2, 0),$
- iii) $\bar{\mathbf{p}} \circledast \mathbf{q} \oplus \bar{\mathbf{q}} \circledast \mathbf{p} = -2(a\alpha + b\beta, 0).$

Demostración. En efecto,

$$\mathbf{p} \circledast \mathbf{q} = (a, b) \circledast (\alpha, \beta) = (-a\alpha + b\beta, a\beta + b\alpha) = (-a\alpha + b\beta, -a\beta - b\alpha)$$

$$= (\alpha, -\beta) \circledast (a, -b) = \bar{\mathbf{q}} \circledast \bar{\mathbf{p}}.$$

También,

$$\mathbf{p} \circledast \bar{\mathbf{p}} = (a, b) \circledast (a, -b) = (-a^2 - b^2, 0) = -(a^2 + b^2, 0).$$

Finalmente, se tiene que:

$$\bar{\mathbf{p}} \circledast \mathbf{q} = (a, -b) \circledast (\alpha, \beta) = (-a\alpha - b\beta, -b\alpha + a\beta),$$

y

$$\bar{\mathbf{q}} \circledast \mathbf{p} = (\alpha, -\beta) \circledast (a, b) = (-\alpha a - \beta b, -\beta a + \alpha b),$$

por tanto,

$$\bar{\mathbf{p}} \circledast \mathbf{q} \oplus \bar{\mathbf{q}} \circledast \mathbf{p} = -2(a\alpha + b\beta, 0).$$

■

Tomando en cuenta la transformación $T_R : \mathcal{C}_R \rightarrow \mathfrak{R}$, definida en incisos anteriores, observamos que la siguiente relación es satisfecha:

$$\langle \mathbf{p}, \mathbf{q} \rangle = -\frac{1}{2} T_R \{ \bar{\mathbf{p}} \circledast \mathbf{q} \oplus \bar{\mathbf{q}} \circledast \mathbf{p} \}.$$

1.5. Otra parametrización de reflexiones.

Se presenta en esta parte, una transformación lineal $\rho_2(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$, con $\mathbf{p} \in \mathfrak{R}^2$ fijo, definida a través de la operación estudiada en el inciso anterior, la cual preserva el producto interno y el determinante asociado a la matriz de la misma, construida con respecto a la base canónica de \mathfrak{R}^2 , resulta ser negativo, con lo cual dicha transformación es una reflexión. Se estudiarán también algunas otras representaciones de la reflexión construida. Sea $\rho_2(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$, con $\mathbf{p} \in \mathfrak{R}^2$ fijo, definida por:

$$\rho_2(\mathbf{p}, \mathbf{q}) = \frac{1}{\|\mathbf{p}\|} \mathbf{p} \circledast \mathbf{q} \quad , \quad \forall \mathbf{q} \in \mathfrak{R}^2 . \quad (1.5)$$

Observemos que la transformación $\rho_2(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ tiene la misma forma cualitativa que la definida en el inciso 1.4. En este caso $\rho_2(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ está definida en términos de la operación $\circledast: \mathfrak{R}^2 \times \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ definida en el inciso 1.5. Algunas propiedades de la misma se estudian a continuación.

Teorema 1.17. La transformación $\rho_2(\mathbf{p}, \bullet): \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ es lineal, ortogonal de determinante negativo.

Demostración. En efecto, siendo $\mathbf{q}, \mathbf{r} \in \mathfrak{R}^2$ y de acuerdo a la distribuidad de la operación $\circledast: \mathfrak{R}^2 \times \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$, se obtiene que :

$$\begin{aligned} \rho_2(\mathbf{p}, \mathbf{q} \oplus \mathbf{r}) &\equiv \frac{1}{\|\mathbf{p}\|} \mathbf{p} \circledast (\mathbf{q} \oplus \mathbf{r}) \equiv \frac{1}{\|\mathbf{p}\|} \{ \mathbf{p} \circledast \mathbf{q} \oplus \mathbf{p} \circledast \mathbf{r} \} \\ &\equiv \frac{1}{\|\mathbf{p}\|} \mathbf{p} \circledast \mathbf{q} \oplus \frac{1}{\|\mathbf{p}\|} \mathbf{p} \circledast \mathbf{r} \equiv \rho_2(\mathbf{p}, \mathbf{q}) \oplus \rho_2(\mathbf{p}, \mathbf{r}). \end{aligned}$$

Además, siendo $\alpha \in \mathbb{R}, \mathbf{q} \in \mathbb{R}^2$

$$\rho_2(\mathbf{p}, \alpha \mathbf{q}) \equiv \frac{1}{\|\mathbf{p}\|} \mathbf{p} \circledast (\alpha \mathbf{q}) \equiv \alpha \frac{1}{\|\mathbf{p}\|} \mathbf{p} \circledast \mathbf{q} \equiv \alpha \rho_2(\mathbf{p}, \mathbf{q}).$$

También, siendo $\mathbf{q} \in \mathbb{R}^2, \mathbf{r} \in \mathbb{R}^2$, entonces,

$$\begin{aligned} \langle \rho_2(\mathbf{p}, \mathbf{q}), \rho_2(\mathbf{p}, \mathbf{r}) \rangle &= -\frac{1}{2} \text{Tr} \{ \overline{\rho_2(\mathbf{p}, \mathbf{q})} \circledast \rho_2(\mathbf{p}, \mathbf{r}) \oplus \overline{\rho_2(\mathbf{p}, \mathbf{r})} \circledast \rho_2(\mathbf{p}, \mathbf{q}) \} \\ &= -\frac{1}{2} \text{Tr} \left\{ \frac{1}{\|\mathbf{p}\|^2} \overline{(\mathbf{p} \circledast \mathbf{q})} \circledast (\mathbf{p} \circledast \mathbf{r}) \oplus \overline{(\mathbf{p} \circledast \mathbf{r})} \circledast (\mathbf{p} \circledast \mathbf{q}) \right\} \\ &= -\frac{1}{2} \frac{1}{\|\mathbf{p}\|^2} \text{Tr} \{ \overline{(\mathbf{p} \circledast \mathbf{q})} \circledast (\mathbf{p} \circledast \mathbf{r}) \oplus \overline{(\mathbf{p} \circledast \mathbf{r})} \circledast (\mathbf{p} \circledast \mathbf{q}) \}. \end{aligned}$$

El resultado buscado, se sigue al desarrollar esta última expresión, esto es, al observar que:

$$(\mathbf{p} \circledast \mathbf{q}) \circledast (\mathbf{p} \circledast \mathbf{r}) = -\|\mathbf{p}\|^2 (\alpha c + \beta d, \beta c - \alpha d),$$

y

$$(\mathbf{p} \circledast \mathbf{r}) \circledast (\mathbf{p} \circledast \mathbf{q}) = -\|\mathbf{p}\|^2 (c\alpha + d\beta, d\alpha - c\beta),$$

de donde,

$$\begin{aligned} (\mathbf{p} \circledast \mathbf{q}) \circledast (\mathbf{p} \circledast \mathbf{r}) \oplus (\mathbf{p} \circledast \mathbf{r}) \circledast (\mathbf{p} \circledast \mathbf{q}) &= -2 \|\mathbf{p}\|^2 (c\alpha + d\beta, 0) \\ &= -2 \|\mathbf{p}\|^2 \text{Tr}^{-1} \{ \langle \mathbf{q}, \mathbf{r} \rangle \}, \end{aligned}$$

por tanto,

$$\langle \rho_2(\mathbf{p}, \mathbf{q}), \rho_2(\mathbf{p}, \mathbf{r}) \rangle \equiv \langle \mathbf{q}, \mathbf{r} \rangle.$$

Finalmente, al observar que:

$$\rho_2(\mathbf{p}, \mathbf{e}_1) = \frac{1}{\|\mathbf{p}\|} (a, b) \otimes (1, 0) = \frac{1}{\|\mathbf{p}\|} (-a, b),$$

y

$$\rho_2(\mathbf{p}, \mathbf{e}_2) = \frac{1}{\|\mathbf{p}\|} (a, b) \otimes (0, 1) = \frac{1}{\|\mathbf{p}\|} (b, a),$$

la matriz de $\rho_2(\mathbf{p}, \bullet) : \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ está dada por:

$$\mathbf{M}_{\rho_2(\mathbf{p}, \bullet)} = \frac{1}{\|\mathbf{p}\|} \begin{pmatrix} -a & b \\ b & a \end{pmatrix},$$

de donde, $\det \mathbf{M}_{\rho_2(\mathbf{p}, \bullet)} \equiv -1$.

■

Teorema 1.18. La transformación $\rho_2(\mathbf{p}, \bullet) : \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$, puede ser representada en la forma siguiente:

$$\rho_2(\mathbf{p}, \bullet) = \frac{1}{\|\mathbf{p}\|} \{a \mathbf{I}_{\sim_2} + b \mathbf{W}_{\sim_2}\} = -\frac{1}{\|\mathbf{p}\|} \{a \mathbf{I}_{\sim_1} + b \mathbf{W}_{\sim_1}\},$$

donde las matrices de las transformación \mathbf{I}_{\sim_2} y \mathbf{W}_{\sim_2} , están dadas por:

$$\mathbf{I}_2 = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \mathbf{W}_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$

y las transformaciones $\tilde{\mathbf{I}}_1$ y $\tilde{\mathbf{W}}_1$ están definidas en el inciso anterior.

Demostración. Es suficiente con observar que la matriz de la transformación estudiada, puede escribirse en la forma siguiente:

$$\begin{aligned} \mathbf{M}_{\rho_2(\mathbf{p}, \bullet)} &= \frac{1}{\|\mathbf{p}\|} \begin{pmatrix} -a & b \\ b & a \end{pmatrix} = \frac{1}{\|\mathbf{p}\|} \left\{ a \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} + b \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \right\} \\ &= \frac{1}{\|\mathbf{p}\|} \left\{ -a \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} - b \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix} \right\}. \end{aligned}$$

■

Teorema 1.19. Sea $\mathbf{q}=(q_0, q_1) \in \mathbb{R}^2$ y $\mathbf{p}=(a, b) \in \mathbb{R}^2$, entonces, las siguientes igualdades son satisfechas.

$$\begin{aligned} \mathbf{q} \times \rho_2(\mathbf{p}, \mathbf{q}) &= \frac{1}{\|\mathbf{p}\|} (b(q_0^2 - q_1^2) + 2a q_0 q_1) \underline{e}_3 \\ &= \frac{1}{\|\mathbf{p}\|} (0, 0, b(q_0^2 - q_1^2) + 2a q_0 q_1), \end{aligned}$$

$$\cos \theta = \frac{1}{\|\mathbf{p}\|} \{-a(q_0^2 - q_1^2) + 2b q_0 q_1\}.$$

Demostración. En efecto,

$$\mathbf{q} \times \rho_2(\mathbf{p}, \mathbf{q}) = \varepsilon_{3jk} q_i \rho_2(\mathbf{p}, \mathbf{q})_k \underline{e}_3 = \{q_0 \rho_2(\mathbf{p}, \mathbf{q})_2 - q_1 \rho_2(\mathbf{p}, \mathbf{q})_1\} \underline{e}_3$$

$$\begin{aligned}
&= \frac{1}{\|\mathbf{p}\|} \{q_0 (aq_1 + bq_0) - q_1 (-aq_0 + bq_1)\} \underline{e}_3, \\
&\equiv \frac{1}{\|\mathbf{p}\|} \{b(q_0^2 - q_1^2) + 2aq_0q_1\} \underline{e}_3 \\
&\equiv \frac{1}{\|\mathbf{p}\|} (0, 0, b(q_1^2 - q_0^2) - 2aq_0q_1).
\end{aligned}$$

Además,

$$\begin{aligned}
\cos \theta &\equiv \langle \rho_2(\mathbf{p}, \mathbf{q}), \mathbf{q} \rangle = \frac{1}{\|\mathbf{p}\|} \langle (-aq_0 + bq_1, aq_1 + bq_0), (q_0, q_1) \rangle \\
&= \frac{1}{\|\mathbf{p}\|} \{-a q_0^2 + bq_0 q_1 + a q_1^2 + bq_1 q_0\} \\
&\equiv \frac{1}{\|\mathbf{p}\|} \{-a(q_0^2 - q_1^2) + 2bq_0q_1\}.
\end{aligned}$$

■

APÉNDICE B
MODELOS DE LA CINEMÁTICA INVERSA

Solución de la cinemática inversa para codo abajo con operación rotación 1 ROT1

```

ClearAll[x, y, L, M, p0, p1, q0, q1, A, B, a, b, c, d, e, f, g, h];
Roa[p_, q_] := {p[[1]]*q[[1]] - p[[2]]*q[[2]], p[[1]]*q[[2]] + p[[2]]*q[[1]]};
u1 = {1, 0};
p = {p0, p1}; q = {q0, q1};
L1 = A; L2 = B;
es1a = L1 * Roa[p, u1];
pnd1 = es1a + L2 * Roa[q, Roa[p, u1]];

sol = Solve[{pnd1[[1]] == x, pnd1[[2]] == y, p0^2 + p1^2 == 1, q0^2 + q1^2 == 1}, {q0, q1, p0, p1}]

```

Solución encontrada:

$$q_0 : \frac{-A^2 - B^2 + x^2 + y^2}{2AB}$$

$$q_1 : \frac{\sqrt{-A^2 x^2 (A^4 + (-B^2 + x^2 + y^2)^2 - 2A^2 (B^2 + x^2 + y^2))}}{2A^2 B x}$$

$$p_0 : \frac{A^3 x^2 + A x^2 (-B^2 + x^2 + y^2) + y \sqrt{-A^2 x^2 (A^4 + (-B^2 + x^2 + y^2)^2 - 2A^2 (B^2 + x^2 + y^2))}}{2A^2 x (x^2 + y^2)}$$

$$p_1 : \frac{A^3 y + A y (-B^2 + x^2 + y^2) - \sqrt{-A^2 x^2 (A^4 + (-B^2 + x^2 + y^2)^2 - 2A^2 (B^2 + x^2 + y^2))}}{2A^2 (x^2 + y^2)}$$

NOTA: Donde x e y son las coordenadas del órgano efector representado por el punto Pot y A , B corresponden a las longitudes de los eslabones L₁ y L₂.

Solución de la cinemática inversa para codo arriba con operación reflexión 2 REF2

```

ClearAll[x, y, L, M, p0, p1, q0, q1, A, B, a, b, c, d, e, f, g, h];
Roa[p_, q_] := {-p[[1]] * q[[1]] + p[[2]] * q[[2]], p[[1]] * q[[2]] + p[[2]] * q[[1]]};
u1 = {1, 0};
p = {p0, p1}; q = {q0, q1};
L1 = A; L2 = B;
es1a = L1 * Roa[p, u1];
pnd1 = es1a + L2 * Roa[q, Roa[p, u1]];

sol = Solve[{pnd1[[1]] = x, pnd1[[2]] = y, p0^2 + p1^2 = 1, q0^2 + q1^2 = 1}, {q0, q1, p0, p1}]
Reflexión : REF2

```

Solución encontrada:

$$\begin{aligned}
 q_0 : & \frac{1}{2A^2B(x^2+y^2)^2} (A^5(x^2-y^2) - A^3(x^2-y^2)(2B^2+x^2+y^2) + AB^2(-x^4+y^4+B^2(x^2-y^2)) - \\
 & 2A^2y\sqrt{-A^2x^2(A^4+(-B^2+x^2+y^2)^2-2A^2(B^2+x^2+y^2))} + \\
 & 2B^2y\sqrt{-A^2x^2(A^4+(-B^2+x^2+y^2)^2-2A^2(B^2+x^2+y^2))}) \\
 q_1 : & \frac{1}{2A^2Bx(x^2+y^2)^2} (-2A^5x^2y + 2AB^2x^2y(-B^2+x^2+y^2) + 2A^3x^2y(2B^2+x^2+y^2) + \\
 & B^2(x^2-y^2)\sqrt{-A^2x^2(A^4+(-B^2+x^2+y^2)^2-2A^2(B^2+x^2+y^2))} + \\
 & A^2(-x^2+y^2)\sqrt{-A^2x^2(A^4+(-B^2+x^2+y^2)^2-2A^2(B^2+x^2+y^2))}) \\
 p_0 : & \left(-\frac{A^3x^2 + Ax^2(-B^2+x^2+y^2) - y\sqrt{-A^2x^2(A^4+(-B^2+x^2+y^2)^2-2A^2(B^2+x^2+y^2))}}{2A^2x(x^2+y^2)} \right) \\
 p_1 : & \frac{A^3y + Ay(-B^2+x^2+y^2) + \sqrt{-A^2x^2(A^4+(-B^2+x^2+y^2)^2-2A^2(B^2+x^2+y^2))}}{2A^2(x^2+y^2)}
 \end{aligned}$$

NOTA: Donde x e y son las coordenadas del órgano efector representado por el punto Pot y A , B corresponden a las longitudes de los eslabones L₁ y L₂.

APÉNDICE C

DIAGRAMAS DE FLUJO PARA LOS CRITERIOS DE EVASIÓN

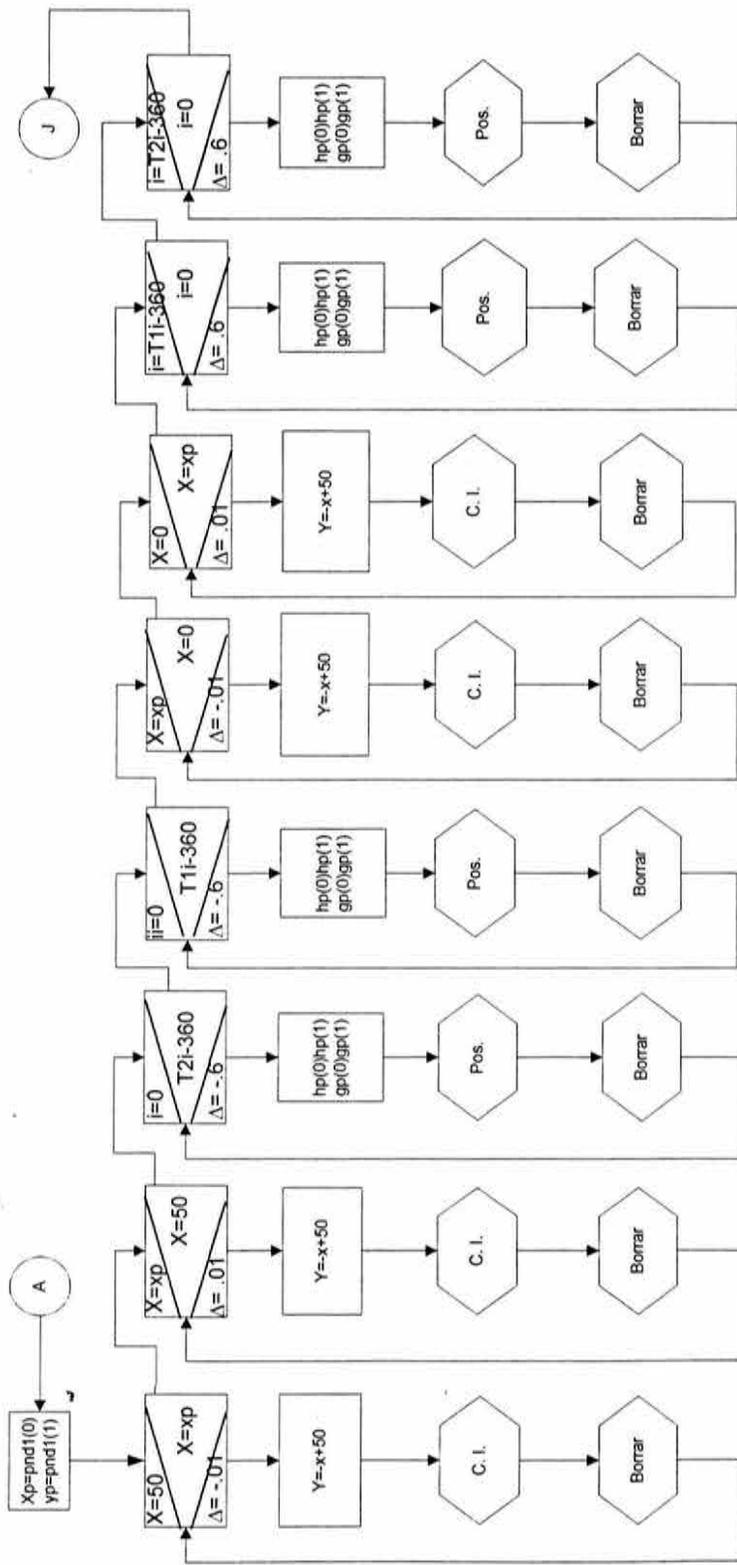


Figura C.1 Diagrama de flujo del Criterio 1

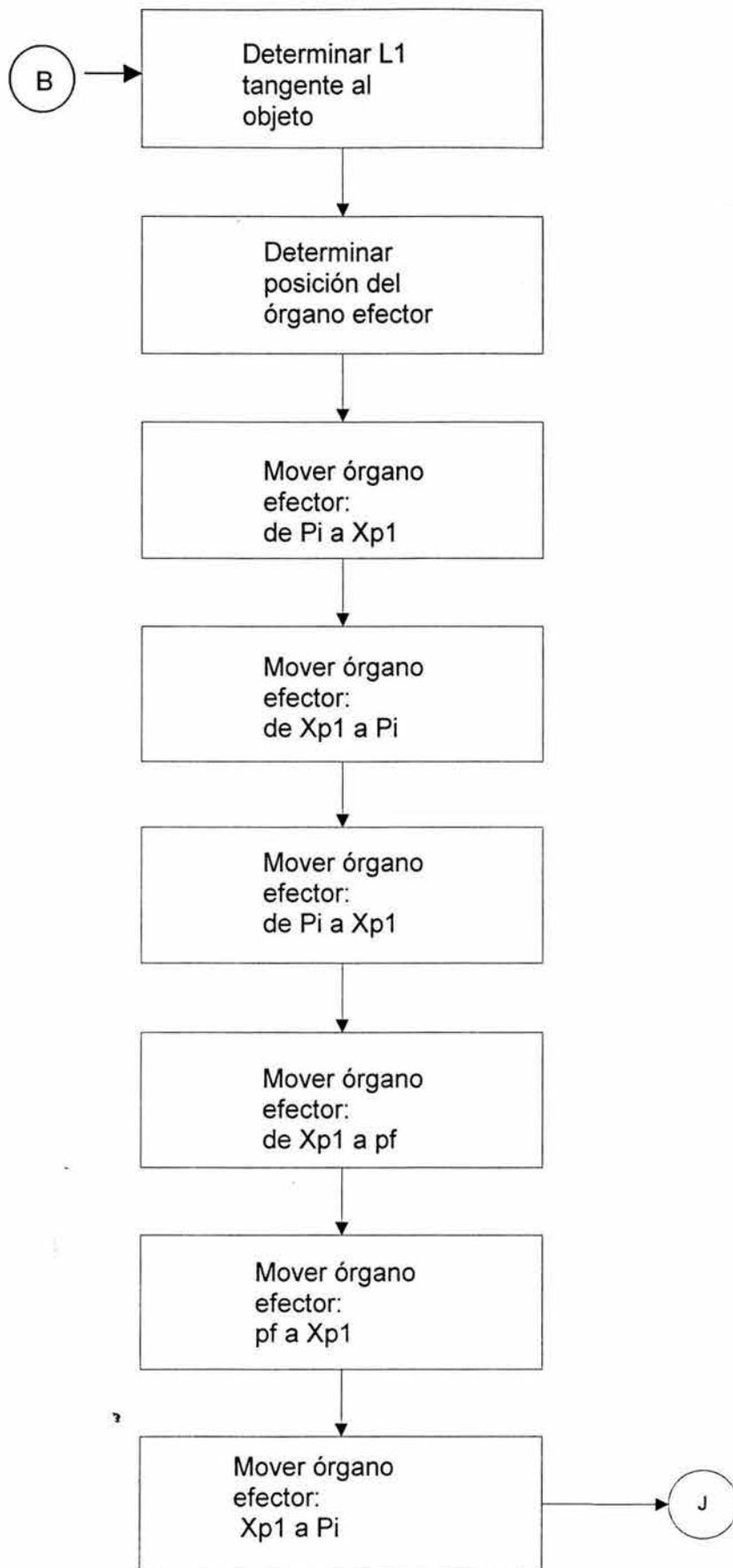


Figura C.2 Diagrama de flujo del Criterio 2

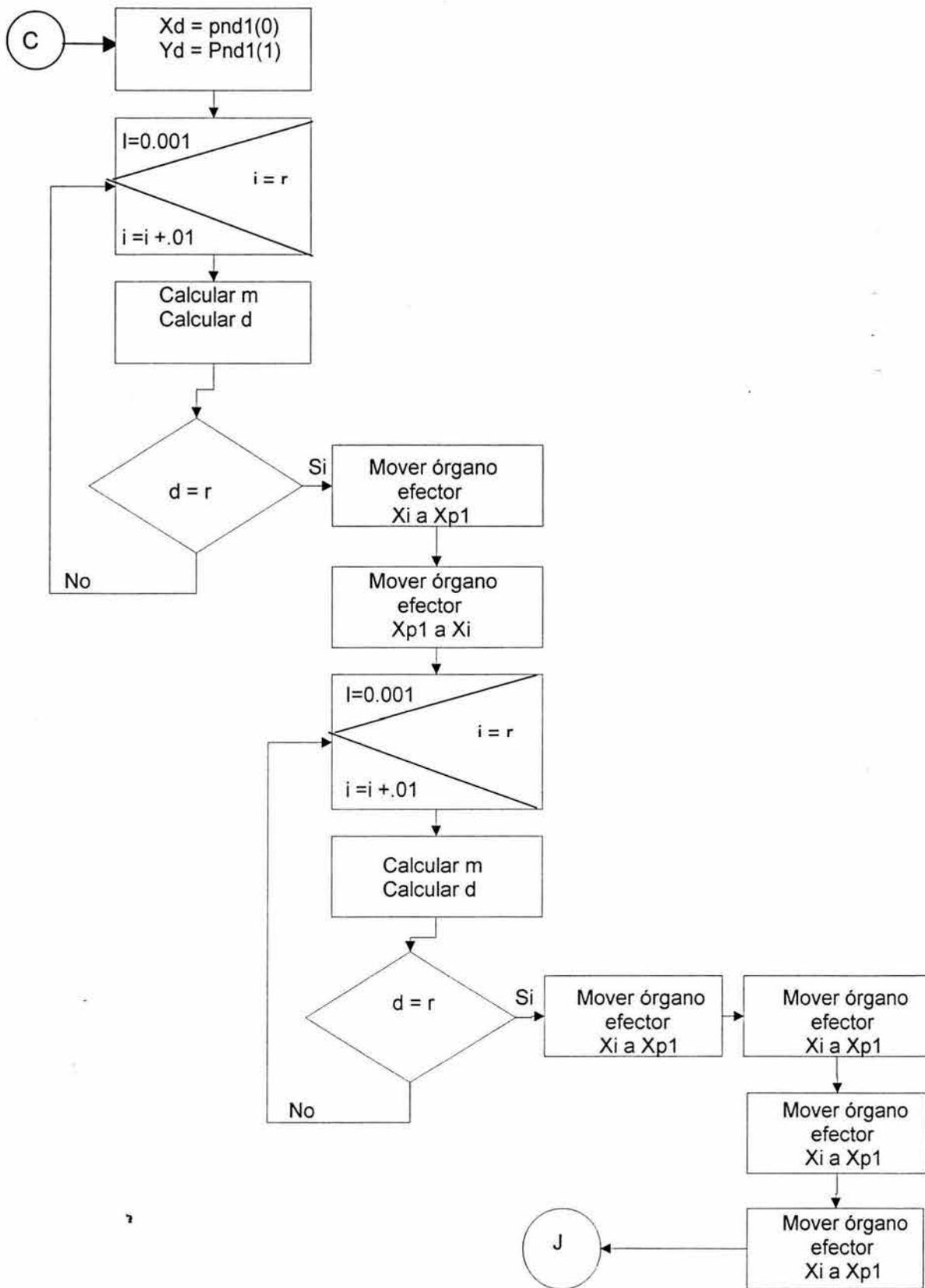


Figura C.3 Diagrama de flujo del Criterio 3

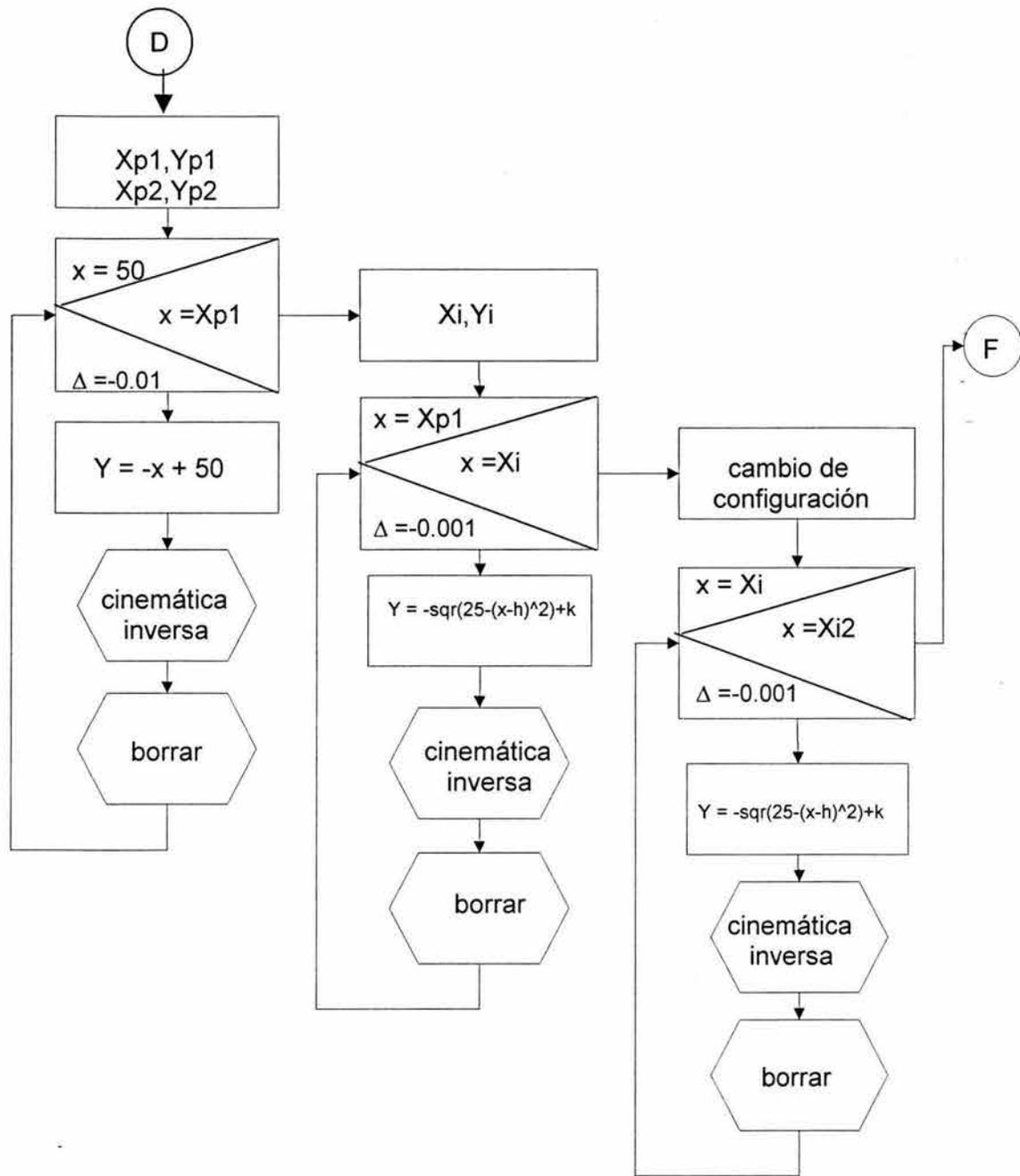


Figura C.4 Diagrama de flujo del Criterio 4

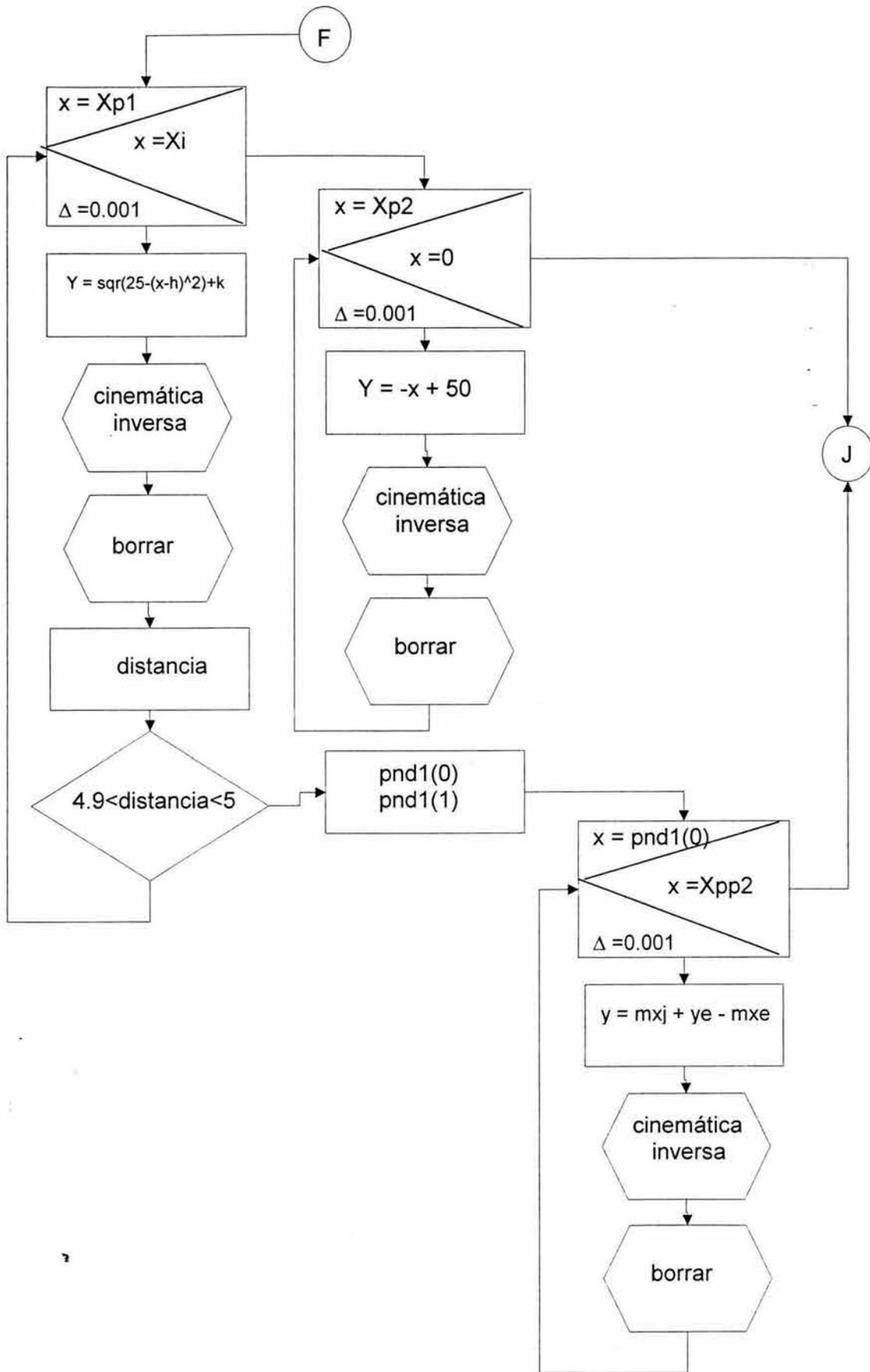


Figura C.5 Diagrama de flujo del Criterio 4

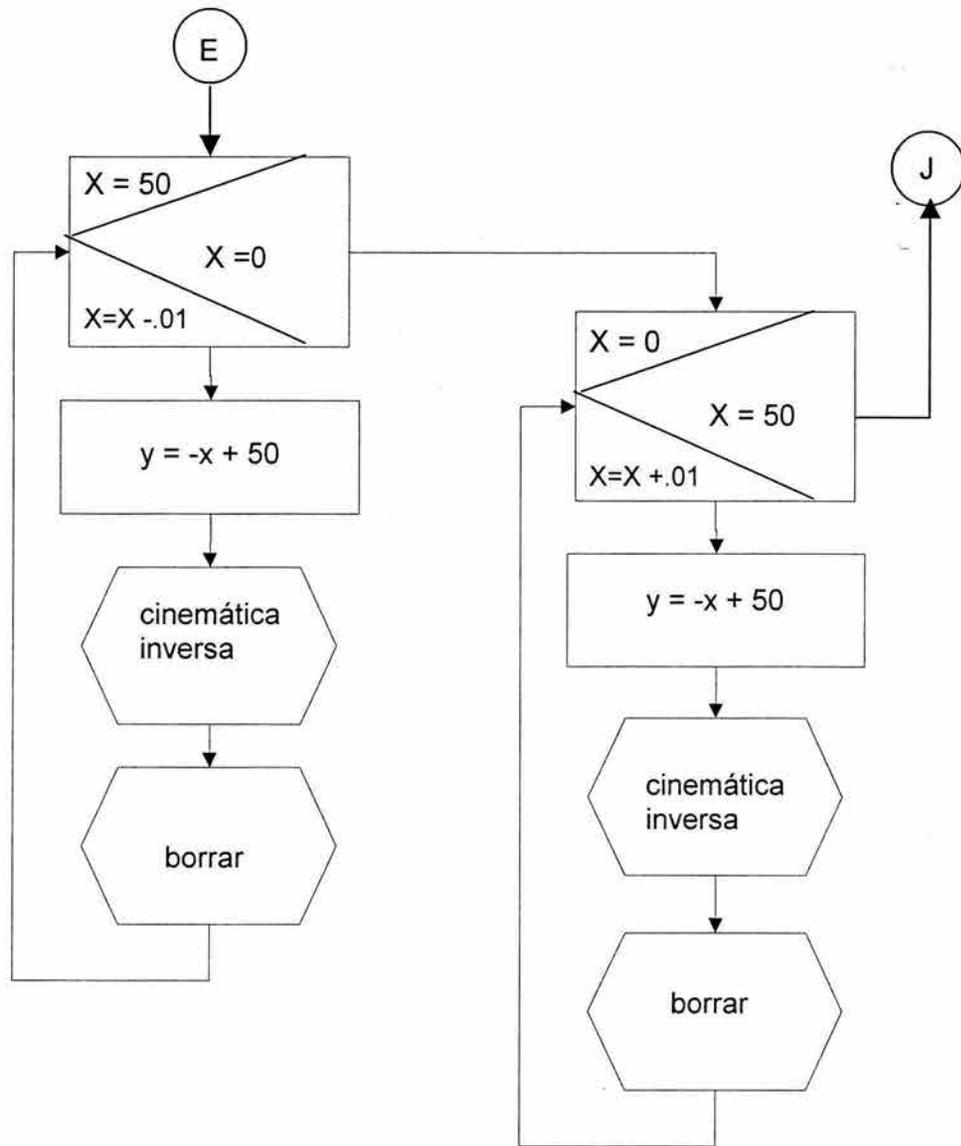


Figura C.6 Diagrama de flujo del Criterio 5

Falta página

N° 194

APÉNDICE D

CÓDIGO DE SECUENCIAS

[INICIA MOVIMIENTOS UNO A UNO]

[Configuración codo arriba]

```
Private Sub Command12_Click()  
    Dim h(1) As Single, g(1) As Single  
  
    If Option5.Value = True Then  
        Picture1.Circle (xi, yi), 0.5, QBColor(5)  
        Picture1.Circle (xf, yf), 0.5, QBColor(5)  
  
    If Option7.Value = True Then  
  
        If theta1_f > theta1_i Then  
            incremento = 0.05  
        Else  
            incremento = -0.05  
        End If
```

[Primer movimiento M1 activo, M2 inactivo]

```
    For i = theta1_i To theta1_f Step incremento  
        h(0) = Cos(i * 3.1416 / 180)  
        h(1) = Sin(i * 3.1416 / 180)  
        g(0) = Cos(theta2_i * 3.1416 / 180)  
        g(1) = Sin(theta2_i * 3.1416 / 180)  
        Call posición(h(0), h(1), g(0), g(1))  
        Call pintar  
    Next i  
    Print theta2_i  
    Call FijarPosición
```

```
    If theta2_f > theta2_i Then  
        incremento = 0.05  
    Else  
        incremento = -0.05  
    End If
```

[Segundo movimiento M1 inactivo, M2 activo]

```
    For i = theta2_i To theta2_f Step incremento  
        h(0) = Cos(theta1_f * 3.1416 / 180)  
        h(1) = Sin(theta1_f * 3.1416 / 180)  
        g(0) = Cos(i * 3.1416 / 180)  
        g(1) = Sin(i * 3.1416 / 180)  
        Call posición(h(0), h(1), g(0), g(1))  
        Call pintar  
    Next i  
  
    Call FijarPosición
```

[A partir de aquí se invierte el orden de la secuencia ahora será primero M2 y luego M1]

Elseif Option9.Value = True Then

[Primer movimiento M1 inactivo, M2 activo]

```
If theta2_f > theta2_i Then
    incremento = 0.05
Else
    incremento = -0.05
End If
For i = theta2_i To theta2_f Step incremento
    h(0) = Cos(theta1_i * 3.1416 / 180)
    h(1) = Sin(theta1_i * 3.1416 / 180)
    g(0) = Cos(i * 3.1416 / 180)
    g(1) = Sin(i * 3.1416 / 180)
    Call posición(h(0), h(1), g(0), g(1))
    Call pintar
Next i

Call FijarPosición
```

[Segundo movimiento M1 activo, M2 inactivo]

```
If theta1_f > theta1_i Then
    incremento = 0.05
Else
    incremento = -0.05
End If
For i = theta1_i To theta1_f Step incremento
    h(0) = Cos(i * 3.1416 / 180)
    h(1) = Sin(i * 3.1416 / 180)
    g(0) = Cos(theta2_f * 3.1416 / 180)
    g(1) = Sin(theta2_f * 3.1416 / 180)
    Call posición(h(0), h(1), g(0), g(1))
    Call pintar
Next i
Print theta2_i
Call FijarPosición
```

Elseif Option10.Value = True Then

[Secuencia -M1/-M2]

```
incremento = -0.05
For i = theta1_i To theta1_f - 360 Step incremento
    h(0) = Cos(i * 3.1416 / 180)
```

```
h(1) = Sin(i * 3.1416 / 180)
g(0) = Cos(theta2_i * 3.1416 / 180)
g(1) = Sin(theta2_i * 3.1416 / 180)
Call posición(h(0), h(1), g(0), g(1))
Call pintar
Next i
```

Call FijarPosición

[M1 inactivo, M2 activo]

```
If theta2_f > theta2_i Then
    incremento = -0.05
    signo = -1
Else
    incremento = 0.05
    signo = 1
End If
```

```
For i = theta2_i To signo * 360 + theta2_f Step incremento
h(0) = Cos(theta1_f * 3.1416 / 180)
h(1) = Sin(theta1_f * 3.1416 / 180)
g(0) = Cos(i * 3.1416 / 180)
g(1) = Sin(i * 3.1416 / 180)
Call posición(h(0), h(1), g(0), g(1))
Call pintar
Next i
```

Call FijarPosición

Elseif Option11.Value = True Then

[Secuencia -M2/-M1]

[M1 inactivo, M2 activo]

```
If theta2_f > theta2_i Then
    incremento = -0.05
    signo = -1
Else
    incremento = 0.05
    signo = 1
End If
```

```
For i = theta2_i To signo * 360 + theta2_f Step incremento
h(0) = Cos(theta1_i * 3.1416 / 180)
h(1) = Sin(theta1_i * 3.1416 / 180)
g(0) = Cos(i * 3.1416 / 180)
```

```
g(1) = Sin(i * 3.1416 / 180)
Call posición(h(0), h(1), g(0), g(1))
Call pintar
Next i
```

```
Call FijarPosición
```

[-M1 activo, M2 inactivo]

```
incremento = -0.05
For i = theta1_i To theta1_f - 360 Step incremento
  h(0) = Cos(i * 3.1416 / 180)
  h(1) = Sin(i * 3.1416 / 180)
  g(0) = Cos(theta2_f * 3.1416 / 180)
  g(1) = Sin(theta2_f * 3.1416 / 180)
  Call posición(h(0), h(1), g(0), g(1))
  Call pintar
Next i
```

```
Call FijarPosición
```

```
Elseif Option12.Value = True Then
```

[Secuencia -M1/M2]

```
incremento = -0.05
For i = theta1_i To theta1_f - 360 Step incremento
  h(0) = Cos(i * 3.1416 / 180)
  h(1) = Sin(i * 3.1416 / 180)
  g(0) = Cos(theta2_i * 3.1416 / 180)
  g(1) = Sin(theta2_i * 3.1416 / 180)
  Call posición(h(0), h(1), g(0), g(1))
  Call pintar
Next i
```

```
Call FijarPosición
```

[Movimiento M1 inactivo, M2 activo]

```
If theta2_f > theta2_i Then
  incremento = 0.05
Else
  incremento = -0.05
End If
For i = theta2_i To theta2_f Step incremento
  h(0) = Cos(theta1_f * 3.1416 / 180)
  h(1) = Sin(theta1_f * 3.1416 / 180)
  g(0) = Cos(i * 3.1416 / 180)
  g(1) = Sin(i * 3.1416 / 180)
  Call posición(h(0), h(1), g(0), g(1))
  Call pintar
Next i
```

Call FijarPosición

Elseif Option13.Value = True Then

'secuencia M1/-M2

If theta1_f > theta1_i Then

 incremento = 0.05

Else

 incremento = -0.05

End If

[Primer movimiento M1 activo, M2 inactivo]

For i = theta1_i To theta1_f Step incremento

h(0) = Cos(i * 3.1416 / 180)

h(1) = Sin(i * 3.1416 / 180)

g(0) = Cos(theta2_i * 3.1416 / 180)

g(1) = Sin(theta2_i * 3.1416 / 180)

Call posición(h(0), h(1), g(0), g(1))

Call pintar

Next i

Print theta2_i

Call FijarPosición

[Segundo movimiento M1 inactivo, M2 activo]

If theta2_f > theta2_i Then

 incremento = -0.05

 signo = -1

Else

 incremento = 0.05

 signo = 1

End If

For i = theta2_i To signo * 360 + theta2_f Step incremento

h(0) = Cos(theta1_f * 3.1416 / 180)

h(1) = Sin(theta1_f * 3.1416 / 180)

g(0) = Cos(i * 3.1416 / 180)

g(1) = Sin(i * 3.1416 / 180)

Call posición(h(0), h(1), g(0), g(1))

Call pintar

Next i

Call FijarPosición

Elseif Option14.Value = True Then

[Secuencia M2/-M1 primer movimiento M1 inactivo, M2 activo]

If theta2_f > theta2_i Then

incremento = 0.05

Else

incremento = -0.05

End If

For i = theta2_i To theta2_f Step incremento

h(0) = Cos(theta1_i * 3.1416 / 180)

h(1) = Sin(theta1_i * 3.1416 / 180)

g(0) = Cos(i * 3.1416 / 180)

g(1) = Sin(i * 3.1416 / 180)

Call posición(h(0), h(1), g(0), g(1))

Call pintar

Next i

Call FijarPosición

incremento = -0.05

For i = theta1_i To theta1_f - 360 Step incremento

h(0) = Cos(i * 3.1416 / 180)

h(1) = Sin(i * 3.1416 / 180)

g(0) = Cos(theta2_f * 3.1416 / 180)

g(1) = Sin(theta2_f * 3.1416 / 180)

Call posición(h(0), h(1), g(0), g(1))

Call pintar

Next i

Call FijarPosición

Elseif Option15.Value = True Then

[M1 inactivo, M2 activo]

If theta2_f > theta2_i Then

incremento = -0.05

signo = -1

Else

incremento = 0.05

signo = 1

End If

For i = theta2_i To signo * 360 + theta2_f Step incremento

h(0) = Cos(theta1_i * 3.1416 / 180)

h(1) = Sin(theta1_i * 3.1416 / 180)

g(0) = Cos(i * 3.1416 / 180)

g(1) = Sin(i * 3.1416 / 180)

Call posición(h(0), h(1), g(0), g(1))

Call pintar

Next i

Call FijarPosición

[Secuencia -M2/M1 primer movimiento M1 activo, M2 inactivo]

```

If theta1_f > theta1_i Then
    incremento = 0.05
Else
    incremento = -0.05
End If

```

```

For i = theta1_i To theta1_f Step incremento
    h(0) = Cos(i * 3.1416 / 180)
    h(1) = Sin(i * 3.1416 / 180)
    g(0) = Cos(theta2_f * 3.1416 / 180)
    g(1) = Sin(theta2_f * 3.1416 / 180)
    Call posición(h(0), h(1), g(0), g(1))
    Call pintar
Next i
Print theta2_i
Call FijarPosición

```

End If

[Configuración de codo abajo]

```

Picture1.Circle (xi, yi), 0.5, QBColor(5)
Picture1.Circle (xf, yf), 0.5, QBColor(5)
Else
    Picture1.Circle (xi, yi), 0.5, QBColor(5)
    Picture1.Circle (xf, yf), 0.5, QBColor(5)
If Option7.Value = True Then

```

```

    If theta1_f > theta1_i Then
        incremento = 0.05
    Else
        incremento = -0.05
    End If

```

[M1 activo, M2 inactivo]

```

For i = theta1_i To theta1_f Step incremento
    h(0) = Cos(i * 3.1416 / 180)
    h(1) = Sin(i * 3.1416 / 180)
    g(0) = Cos(theta2_i * 3.1416 / 180)
    g(1) = Sin(theta2_i * 3.1416 / 180)
    Call posición(h(0), h(1), g(0), g(1))
    Call pintar
Next i
Print theta2_i
Call FijarPosición

```

```

If theta2_f > theta2_i Then
    incremento = 0.05

```

```
Else
  incremento = -0.05
End If
```

[M1 inactivo, M2 activo]

```
For i = theta2_i To theta2_f Step incremento
  h(0) = Cos(theta1_f * 3.1416 / 180)
  h(1) = Sin(theta1_f * 3.1416 / 180)
  g(0) = Cos(i * 3.1416 / 180)
  g(1) = Sin(i * 3.1416 / 180)
  Call posición(h(0), h(1), g(0), g(1))
  Call pintar
Next i
```

Call FijarPosición

[A partir de aquí se invierte el orden de la secuencia ahora será primero M2 y luego M1]

```
Elseif Option9.Value = True Then
```

[Primer movimiento M1 inactivo, M2 activo]

```
If theta2_f > theta2_i Then
  incremento = 0.05
Else
  incremento = -0.05
End If
For i = theta2_i To theta2_f Step incremento
  h(0) = Cos(theta1_i * 3.1416 / 180)
  h(1) = Sin(theta1_i * 3.1416 / 180)
  g(0) = Cos(i * 3.1416 / 180)
  g(1) = Sin(i * 3.1416 / 180)
  Call posición(h(0), h(1), g(0), g(1))
  Call pintar
Next i
```

Call FijarPosición

[Segundo movimiento M1 activo, M2 inactivo]

```
If theta1_f > theta1_i Then
  incremento = 0.05
Else
  incremento = -0.05
End If
For i = theta1_i To theta1_f Step incremento
  h(0) = Cos(i * 3.1416 / 180)
```

```

h(1) = Sin(i * 3.1416 / 180)
g(0) = Cos(theta2_f * 3.1416 / 180)
g(1) = Sin(theta2_f * 3.1416 / 180)
Call posición(h(0), h(1), g(0), g(1))
Call pintar
Next i
Print theta2_i
Call FijarPosición

```

Elseif Option10.Value = True Then

[Secuencia -M1/-M2]

```

incremento = -0.05
For i = theta1_i To theta1_f - 360 Step incremento
  h(0) = Cos(i * 3.1416 / 180)
  h(1) = Sin(i * 3.1416 / 180)
  g(0) = Cos(theta2_i * 3.1416 / 180)
  g(1) = Sin(theta2_i * 3.1416 / 180)
  Call posición(h(0), h(1), g(0), g(1))
  Call pintar
Next i

Call FijarPosición

```

[M1 inactivo, M2 activo]

```

If theta2_f > theta2_i Then
  incremento = -0.05
  signo = -1
Else
  incremento = 0.05
  signo = 1
End If

```

```

For i = theta2_i To signo * 360 + theta2_f Step incremento
  h(0) = Cos(theta1_f * 3.1416 / 180)
  h(1) = Sin(theta1_f * 3.1416 / 180)
  g(0) = Cos(i * 3.1416 / 180)
  g(1) = Sin(i * 3.1416 / 180)
  Call posición(h(0), h(1), g(0), g(1))
  Call pintar
Next i

Call FijarPosición

```

Elseif Option11.Value = True Then

[Secuencia -M2/-M1]

[M1 inactivo, M2 activo]

```
If theta2_f > theta2_i Then
    incremento = -0.05
    signo = -1
Else
    incremento = 0.05
    signo = 1
End If
```

```
For i = theta2_i To signo * 360 + theta2_f Step incremento
    h(0) = Cos(theta1_i * 3.1416 / 180)
    h(1) = Sin(theta1_i * 3.1416 / 180)
    g(0) = Cos(i * 3.1416 / 180)
    g(1) = Sin(i * 3.1416 / 180)
    Call posición(h(0), h(1), g(0), g(1))
    Call pintar
Next i
```

Call FijarPosición

[-M1 activo, M2 inactivo]

```
incremento = -0.05
For i = theta1_i To theta1_f - 360 Step incremento
    h(0) = Cos(i * 3.1416 / 180)
    h(1) = Sin(i * 3.1416 / 180)
    g(0) = Cos(theta2_f * 3.1416 / 180)
    g(1) = Sin(theta2_f * 3.1416 / 180)
    Call posición(h(0), h(1), g(0), g(1))
    Call pintar
Next i
```

Call FijarPosición

Elseif Option12.Value = True Then

[Secuencia -M1/M2]

```
incremento = -0.05
For i = theta1_i To theta1_f - 360 Step incremento
    h(0) = Cos(i * 3.1416 / 180)
    h(1) = Sin(i * 3.1416 / 180)
    g(0) = Cos(theta2_i * 3.1416 / 180)
    g(1) = Sin(theta2_i * 3.1416 / 180)
    Call posición(h(0), h(1), g(0), g(1))
    Call pintar
Next i
```

Call FijarPosición

[Segundo movimiento M1 inactivo, M2 activo]

```
If theta2_f > theta2_i Then
  incremento = 0.05
Else
  incremento = -0.05
End If
For i = theta2_i To theta2_f Step incremento
  h(0) = Cos(theta1_f * 3.1416 / 180)
  h(1) = Sin(theta1_f * 3.1416 / 180)
  g(0) = Cos(i * 3.1416 / 180)
  g(1) = Sin(i * 3.1416 / 180)
  Call posición(h(0), h(1), g(0), g(1))
  Call pintar
Next i
```

Call FijarPosición

```
Elseif Option13.Value = True Then
'secuencia M1/-M2
  If theta1_f > theta1_i Then
    incremento = 0.05
  Else
    incremento = -0.05
  End If
```

[M1 activo, M2 inactivo]

```
For i = theta1_i To theta1_f Step incremento
  h(0) = Cos(i * 3.1416 / 180)
  h(1) = Sin(i * 3.1416 / 180)
  g(0) = Cos(theta2_i * 3.1416 / 180)
  g(1) = Sin(theta2_i * 3.1416 / 180)
  Call posición(h(0), h(1), g(0), g(1))
  Call pintar
Next i
Print theta2_i
Call FijarPosición
```

[M1 inactivo, M2 activo]

```
If theta2_f > theta2_i Then
  incremento = -0.05
  signo = -1
Else
  incremento = 0.05
  signo = 1
End If
```

```

For i = theta2_i To signo * 360 + theta2_f Step incremento
h(0) = Cos(theta1_f * 3.1416 / 180)
h(1) = Sin(theta1_f * 3.1416 / 180)
g(0) = Cos(i * 3.1416 / 180)
g(1) = Sin(i * 3.1416 / 180)
Call posición(h(0), h(1), g(0), g(1))
Call pintar
Next i

```

Call FijarPosición

Elseif Option14.Value = True Then

[Secuencia M2/-M1, M1 inactivo, M2 activo]

```

If theta2_f > theta2_i Then
    incremento = 0.05
Else
    incremento = -0.05
End If
For i = theta2_i To theta2_f Step incremento
h(0) = Cos(theta1_i * 3.1416 / 180)
h(1) = Sin(theta1_i * 3.1416 / 180)
g(0) = Cos(i * 3.1416 / 180)
g(1) = Sin(i * 3.1416 / 180)
Call posición(h(0), h(1), g(0), g(1))
Call pintar
Next i

```

Call FijarPosición

```

incremento = -0.05
For i = theta1_i To theta1_f - 360 Step incremento
h(0) = Cos(i * 3.1416 / 180)
h(1) = Sin(i * 3.1416 / 180)
g(0) = Cos(theta2_f * 3.1416 / 180)
g(1) = Sin(theta2_f * 3.1416 / 180)
Call posición(h(0), h(1), g(0), g(1))
Call pintar
Next i

```

Call FijarPosición

Elseif Option15.Value = True Then

[M1 inactivo, M2 activo]

```

If theta2_f > theta2_i Then
    incremento = -0.05
    signo = -1

```

```

Else
    incremento = 0.05
    signo = 1
End If

```

```

For i = theta2_i To signo * 360 + theta2_f Step incremento
h(0) = Cos(theta1_i * 3.1416 / 180)
h(1) = Sin(theta1_i * 3.1416 / 180)
g(0) = Cos(i * 3.1416 / 180)
g(1) = Sin(i * 3.1416 / 180)
Call posición(h(0), h(1), g(0), g(1))
Call pintar
Next i

```

```

Call FijarPosición

```

[Secuencia -M2/M1 primer movimiento M1 activo, M2 inactivo]

```

If theta1_f > theta1_i Then
    incremento = 0.05
Else
    incremento = -0.05
End If

```

[M1 activo, M2 inactivo]

```

For i = theta1_i To theta1_f Step incremento
h(0) = Cos(i * 3.1416 / 180)
h(1) = Sin(i * 3.1416 / 180)
g(0) = Cos(theta2_f * 3.1416 / 180)
g(1) = Sin(theta2_f * 3.1416 / 180)
Call posición(h(0), h(1), g(0), g(1))
Call pintar
Next i
Print theta2_i
Call FijarPosición

```

```

End If
Picture1.Circle (xi, yi), 0.5, QBColor(5)
Picture1.Circle (xf, yf), 0.5, QBColor(5)
End If

```

```

If Option6.Value = True Then
    pnd1p(0) = pnd1(0)
    pnd1p(1) = pnd1(1)
End If

```

```

End Sub

```

[TERMINA MOVIMIENTOS UNO A UNO]

[INICIA MOVIMIENTOS SIMULTÁNEOS]

```
Private Sub Command2_Click()
```

```
Dim h(1) As Single, g(1) As Single
```

```
If Option1.Value = True Then
```

[Activar simultáneamente el Actuador 1 y 2 en T1, y en T2 solo el Actuador 2]

```
Po = Int(Val(Text6))
```

```
Text11.Text = Po - 360
```

```
For i = 0 To Po Step 0.1
```

```
    ' Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(9)
```

```
    h(0) = Cos(i * 3.1416 / 180)
```

```
    h(1) = Sin(i * 3.1416 / 180)
```

```
    g(0) = Cos((i) * 3.1416 / 180)
```

```
    g(1) = Sin((i) * 3.1416 / 180)
```

```
    Call posición(h(0), h(1), g(0), g(1))
```

```
    Call pintar
```

```
Next
```

```
Call FijarPosición
```

```
Elseif Option2.Value = True Then
```

[Activar simultáneos donde Po > Qo]

```
Po = Int(Val(Text9))
```

```
Qo = Int(Val(Text10))
```

```
Text14.Text = Po - 360
```

```
Text15.Text = Qo - 360
```

```
diferencia = (Po) * 10 / Qo
```

```
cont = 0: cont2 = 0
```

```
For i = 0 To Po Step 0.1
```

```
    'Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(9)
```

```
    h(0) = Cos(i * 3.1416 / 180)
```

```
    h(1) = Sin(i * 3.1416 / 180)
```

```
    cont = cont + 1
```

```
If (cont >= diferencia) And (cont Mod diferencia = 0) Then
```

```
    cont2 = cont2 + 1
```

```
    g(0) = Cos(((cont2)) * 3.1416 / 180)
```

```
    g(1) = Sin(((cont2)) * 3.1416 / 180)
```

```
End If
```

```
Call posición(h(0), h(1), g(0), g(1))
```

```
Call pintar
```

```
Next
```

```
Print cont; cont2; diferencia
```

```
Call FijarPosición
```

Elseif Option3.Value = True Then

[Activar simultáneos donde Po < Qo]

```
Po = Int(Val(Text7))
Qo = Int(Val(Text8))
Text12.Text = Po - 360
Text13.Text = Qo - 360
diferencia = (Qo) * 10 / Po
cont = 0: cont2 = 0
```

```
For i = 0 To Qo Step 0.1
```

```
    'Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(9)
    g(0) = Cos(i * 3.1416 / 180)
    g(1) = Sin(i * 3.1416 / 180)
    cont = cont + 1
```

```
    If (cont >= diferencia) And (cont Mod diferencia = 0) Then
```

```
        cont2 = cont2 + 1
        h(0) = Cos(((cont2)) * 3.1416 / 180)
        h(1) = Sin(((cont2)) * 3.1416 / 180)
```

```
    End If
```

```
    Call posición(h(0), h(1), g(0), g(1))
```

```
    Call pintar
```

```
Next
```

```
Call FijarPosición
```

Elseif Option4.Value = True Then

[Activar simultáneos donde Po* = Qo*]

```
Po = Int(Val(Text11))
```

```
For i = 0 To Po Step -0.1
```

```
    'Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(9)
```

```
    h(0) = Cos(i * 3.1416 / 180)
```

```
    h(1) = Sin(i * 3.1416 / 180)
```

```
    g(0) = Cos(((i)) * 3.1416 / 180)
```

```
    g(1) = Sin((i) * 3.1416 / 180)
```

```
    Call posición(h(0), h(1), g(0), g(1))
```

```
    Call pintar
```

```
Next
```

```
Call FijarPosición
```

Elseif Option16.Value = True Then

[Activar simultáneos donde Po* > Qo*]

```
Po = Int(Val(Text12))
Qo = Int(Val(Text13))
diferencia = Abs(Po) * 10 / Abs(Qo)
cont = 0: cont2 = 0
```

```
For i = 0 To Po Step -0.1
```

```
  'Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(9)
  h(0) = Cos(i * 3.1416 / 180)
  h(1) = Sin(i * 3.1416 / 180)
  cont = cont + 1
```

```
If ((cont >= diferencia) And (cont Mod diferencia = 0)) And (cont2 <= Abs(Qo)) Then
```

```
  cont2 = cont2 + 1
  g(0) = Cos(-cont2 * 3.1416 / 180)
  g(1) = Sin(-cont2 * 3.1416 / 180)
```

```
End If
```

```
Call posición(h(0), h(1), g(0), g(1))
```

```
Call pintar
```

```
Next
```

```
If cont2 < Abs(Qo) Then
```

```
  For j = -cont2 To Qo Step -0.01
```

```
    g(0) = Cos(j * 3.1416 / 180)
```

```
    g(1) = Sin(j * 3.1416 / 180)
```

```
    Call posición(h(0), h(1), g(0), g(1))
```

```
    Call pintar
```

```
  Next
```

```
  Print "CONT2 abs Po"; cont2; Abs(Po)
```

```
End If
```

```
Call FijarPosición
```

```
Elseif Option17.Value = True Then
```

[Activar simultáneos donde Po* < Qo*]

```
Po = Int(Val(Text14))
Qo = Int(Val(Text15))
diferencia = Qo * 10 / Po
cont = 0: cont2 = 0
Print "Qo Po"; Qo; Po
```

```
For i = 0 To Qo Step -0.1
```

```
  'Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(9)
```

```
  g(0) = Cos(i * 3.1416 / 180)
```

```
  g(1) = Sin(i * 3.1416 / 180)
```

```
cont = cont + 1
```

```
If (cont >= diferencia) And (cont Mod diferencia = 0) And (cont2 <= Abs(Po)) Then  
    cont2 = cont2 + 1  
    h(0) = Cos(-cont2 * 3.1416 / 180)  
    h(1) = Sin(-cont2 * 3.1416 / 180)  
End If
```

```
Call posición(h(0), h(1), g(0), g(1))  
Call pintar
```

```
Next
```

```
If cont2 < Abs(Po) Then  
    For j = -cont2 To Po Step -0.1  
        h(0) = Cos(j * 3.1416 / 180)  
        h(1) = Sin(j * 3.1416 / 180)  
        Call posición(h(0), h(1), g(0), g(1))  
        Call pintar  
    Next  
    Print "CONT2 abs Po"; cont2; Abs(Po)  
End If
```

```
Call FijarPosición
```

```
Elseif Option18.Value = True Then
```

```
[ Activar simultáneos donde Po* > Qo]
```

```
Po = Int(Val(Text16))  
Qo = Int(Val(Text17))  
diferencia = Abs(Po) * 10 / Qo  
cont = 0: cont2 = 0  
Print "Qo Po"; Qo; Po
```

```
For i = 0 To Po Step -0.1  
    'Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(9)  
    h(0) = Cos(i * 3.1416 / 180)  
    h(1) = Sin(i * 3.1416 / 180)  
    cont = cont + 1
```

```
If (cont >= diferencia) And (cont Mod diferencia = 0) Then  
    cont2 = cont2 + 1  
    g(0) = Cos(cont2 * 3.1416 / 180)  
    g(1) = Sin(cont2 * 3.1416 / 180)  
End If
```

```
Call posición(h(0), h(1), g(0), g(1))
```

Call pintar

Next

```
If cont2 < Abs(Qo) Then
  For j = cont2 To Qo Step 0.1
    g(0) = Cos(j * 3.1416 / 180)
    g(1) = Sin(j * 3.1416 / 180)
    Call posición(h(0), h(1), g(0), g(1))
    Call pintar
  Next
  Print "CONT2 abs Po"; cont2; Abs(Po)
End If
```

Call FijarPosición

Elseif Option19.Value = True Then

[Activar simultáneos donde Qo* > Po]

```
Po = Int(Val(Text18))
Qo = Int(Val(Text19))
diferencia = Abs(Qo) * 10 / Po
cont = 0: cont2 = 0
Print "Qo Po"; Qo; Po
```

```
For i = 0 To Qo Step -0.1
  'Picture1.Circle (pnd1(0), pnd1(1)), 0.1, QBColor(9)
  g(0) = Cos(i * 3.1416 / 180)
  g(1) = Sin(i * 3.1416 / 180)
  cont = cont + 1

  If (cont >= diferencia) And (cont Mod diferencia = 0) Then
    cont2 = cont2 + 1
    h(0) = Cos(cont2 * 3.1416 / 180)
    h(1) = Sin(cont2 * 3.1416 / 180)
  End If
  Call posición(h(0), h(1), g(0), g(1))
  Call pintar
```

Next

```
If cont2 < Abs(Po) Then
  For j = cont2 To Po Step -0.1
    h(0) = Cos(j * 3.1416 / 180)
    h(1) = Sin(j * 3.1416 / 180)
    Call posición(h(0), h(1), g(0), g(1))
    Call pintar
  Next
  Print "CONT2 abs Po"; cont2; Abs(Po)
```

End If

Call FijarPosición

[Terminan los 8 botones de opción en codo arriba]

End IF

End Sub

[TERMINA MOVIMIENTOS SIMULTÁNEOS]

[En el siguiente código se ejemplifican solo tres casos de los movimientos híbridos]

[INICIA MOVIMIENTOS HÍBRIDOS]

Private Sub Command3_Click()

Dim h(1) As Single, g(1) As Single

If Option8.Value = True Then

[En Po y Qo se capturan los parámetros de rotación]

Po = Int(Val(Text22))

Qo = Int(Val(Text23))

[En este ciclo ambos actuadores realizan el mismo desplazamiento Po]

For i = 0 To Po Step 0.1

h(0) = Cos(i * 3.1416 / 180)

h(1) = Sin(i * 3.1416 / 180)

g(0) = Cos((i) * 3.1416 / 180)

g(1) = Sin((i) * 3.1416 / 180)

Call posición(h(0), h(1), g(0), g(1))

Call pintar

Next

[Luego se corrige el parámetro Qo]

If Po < Qo Then

For i = Po To Qo Step 0.1

g(0) = Cos((i) * 3.1416 / 180)

g(1) = Sin((i) * 3.1416 / 180)

Call posición(h(0), h(1), g(0), g(1))

Call pintar

Next

Elseif Po > Qo Then

For i = Po To Qo Step -0.1

g(0) = Cos((i) * 3.1416 / 180)

g(1) = Sin((i) * 3.1416 / 180)

```
    Call posición(h(0), h(1), g(0), g(1))
    Call pintar
  Next
End If
Call FijarPosición
```

```
Elseif Option20.Value = True Then
```

```
  Po = Int(Val(Text24))
  Qo = Int(Val(Text25))
```

[En este ciclo ambos actuadores realizan el mismo desplazamiento Qo]

```
For i = 0 To Qo Step 0.3
  h(0) = Cos(i * 3.1416 / 180)
  h(1) = Sin(i * 3.1416 / 180)
  g(0) = Cos((i) * 3.1416 / 180)
  g(1) = Sin((i) * 3.1416 / 180)
  Call posición(h(0), h(1), g(0), g(1))
  Call pintar
Next
```

[Luego se corrige el parámetro Po]

```
If Po < Qo Then
  For i = Qo To Po Step -0.3
    h(0) = Cos(i * 3.1416 / 180)
    h(1) = Sin(i * 3.1416 / 180)
    g(0) = Cos((Qo) * 3.1416 / 180)
    g(1) = Sin((Qo) * 3.1416 / 180)
    Call posición(h(0), h(1), g(0), g(1))
    Call pintar
  Next
```

```
Elseif Po > Qo Then
  For i = Qo To Po Step 0.3
    h(0) = Cos(i * 3.1416 / 180)
    h(1) = Sin(i * 3.1416 / 180)
    g(0) = Cos((Qo) * 3.1416 / 180)
    g(1) = Sin((Qo) * 3.1416 / 180)
    Call posición(h(0), h(1), g(0), g(1))
    Call pintar
  Next
End If
Call FijarPosición
```

```
Elseif Option21.Value = True Then
```

```
Po = Int(Val(Text26))
Qo = Int(Val(Text27))
Poc = Po - 360
Qoc = Qo - 360
```

[En este ciclo ambos actuadores realizan el mismo desplazamiento Poc, que es el complemento (diferencia de Po y 360°) del parámetro Po]

```
For i = 0 To Poc Step -0.3
  h(0) = Cos(i * 3.1416 / 180)
  h(1) = Sin(i * 3.1416 / 180)
  g(0) = Cos((i) * 3.1416 / 180)
  g(1) = Sin((i) * 3.1416 / 180)
  Call posición(h(0), h(1), g(0), g(1))
  Call pintar
Next
```

```
If Abs(Poc) < Abs(Qoc) Then
  For i = Poc To Qo Step 0.3
    g(0) = Cos((i) * 3.1416 / 180)
    g(1) = Sin((i) * 3.1416 / 180)
    Call posición(h(0), h(1), g(0), g(1))
    Call pintar
  Next
```

```
Elseif Abs(Poc) > Abs(Qoc) Then
  For i = Po To -360 + Qo Step -0.3
    g(0) = Cos((i) * 3.1416 / 180)
    g(1) = Sin((i) * 3.1416 / 180)
    Call posición(h(0), h(1), g(0), g(1))
    Call pintar
  Next
End If
```

```
Call FijarPosición
```