

UNIVERSIDAD DE SALAMANCA
DEPARTAMENTO DE INFORMÁTICA Y AUTOMÁTICA
FACULTAD DE CIENCIAS



FORMALISMO MATEMÁTICO PARA LA
REPRESENTACIÓN DE OBSTÁCULOS
EN EL ESPACIO DE LAS
CONFIGURACIONES DE UN ROBOT

Belén Curto Diego

Junio 1998

D. **Eladio Sanz García**, Catedrático de Ingeniería de Sistemas y Automática de la Universidad de Salamanca y D. **Vidal Moreno Rodilla**, Profesor Titular de Ingeniería de Sistemas y Automática de la Universidad de Salamanca

CERTIFICAN:

Que el trabajo de investigación que se recoge en la presente memoria, titulada **Formalismo Matemático para la Representación de los Obstáculos en el Espacio de las Configuraciones de un Robot** y presentada por Doña **Belén Curto Diego** para optar al grado de Doctor en Ciencias Físicas, ha sido realizado bajo su dirección en el Area de Ingeniería de Sistemas y Automática del Departamento de Informática y Automática de la Universidad de Salamanca.

Salamanca, 8 de Junio de 1998

D. Eladio Sanz García
Catedrático de Ingeniería de
Sistemas y Automática
Universidad de Salamanca

D. Vidal Moreno Rodilla
Profesor Titular de Ingeniería de
Sistemas y Automática
Universidad de Salamanca

A mi madre por sus desvelos

Agradecimientos

Deseo expresar en esta página mi agradecimiento a todas las personas que me han ayudado de forma tanto técnica como moral en la realización de este trabajo de investigación. En primer lugar pido disculpas si cometo el error de dejar a alguien en el olvido o por si no le agradezco suficientemente el esfuerzo que me ha prestado.

Me gustaría dar las gracias a mis directores de tesis, E. Sanz García y V. Moreno Rodilla, por su inestimable apoyo y confianza y los sabios consejos que me han dado.

De manera muy especial me gustaría agradecer a mi compañero Vidal su iniciativa, sus ideas, sus comentarios, su apoyo incondicional, su optimismo ante la vida y el haber hecho todo lo posible en todo momento porque los proyectos que tenemos juntos sigan adelante.

También deseo expresar mi agradecimiento a todas las personas que forman parte del Departamento de Informática y Automática: a Pastora y Belén P. por sus lecturas críticas, sus sugerencias y su paciencia al leer esta memoria; a Nines e Iván por su soporte técnico e informático; a Javi y Roberto por estropear y arreglar inmediatamente el software y el hardware que he utilizado; a Luis Q., Luis A., Angel, María, Juan, Coti, Quique, Angélica y Juanan por los buenos ratos que pasamos juntos y por todo el compañerismo y amistad demostrados; y en general a todos ellos porque hacen del Departamento un entorno muy agradable de trabajo.

Quiero agradecer a Mercedes sus animadas charlas técnicas y personales y su amistad; al profesor Juan Mateos Guilarte por hacerme recordar los conocimientos de la Física aprendidos durante la carrera y su aplicación a este trabajo de investigación.

A mis inseparables compañeros de carrera y de correrías, Sonsoles, Paco y M. José, también quiero incorporarlos a este pequeño apartado. A mis amigas, Inma, Magdalena, Juani, Francis, Ana y M. Angeles, les quiero agradecer que a pesar de la falta de tiempo y de los avatares de la vida, lo sigan siendo.

En último lugar, pero no por ello el menos importante, quiero agradecer a mi familia y, en especial, a mis padres, a mis hermanos y a mi marido, por afrontar la ardua tarea de lidiar con mi carácter y por su apoyo y paciencia infinita; y a mis hijos porque aunque han retrasado la realización de este trabajo son la fuente de mi vida.

Índice general

Capítulo 1

Introducción

La tendencia actual de la Robótica es dotar a los robots de un alto grado de autonomía. Así, el usuario le describirá qué tarea debe realizar en lugar de cómo llevarla a cabo. Los robots autónomos son especialmente interesantes para diversas aplicaciones como agricultura, construcción, ayuda a discapacitados, fabricación, exploración espacial y submarina, etc. En estas aplicaciones no existe un conocimiento detallado del entorno, las tareas son poco repetitivas e incluso se realizan en entornos peligrosos o en los que no es posible el acceso a personas (ambientes contaminados). Estas características obligan a utilizar robots controlados de forma remota o robots que cuenten con un mayor grado de *inteligencia*.

En los sectores donde el robot se ha adaptado de forma óptima, como la industria del automóvil o la de fabricación, éste realiza tareas repetitivas (soldadura, paletización, montaje) ejecutando secuencias de movimientos que previamente le ha programado un operador. Estas operaciones las realiza en un entorno de trabajo estructurado ([?]). Incluso para su implantación en estos procesos se está tratando de dotarle de la capacidad de generar de forma *inteligente* y automática estas secuencias de movimientos.

La evolución hacia robots autónomos supone dejar a un lado el clásico método de aprendizaje y repetición de secuencias de movimiento e implantar otro, donde los robots serán capaces de aceptar descripciones de alto nivel sobre las tareas que deben realizar y ejecutarlas sin la intervención de operadores. En este

progreso están implicadas diferentes disciplinas como la Inteligencia Artificial, las Ciencias de la Computación, el Control Automático, la Tecnología de los Computadores y la Ingeniería Mecánica. Sin embargo, estas disciplinas no están lo suficientemente desarrolladas como para resolver las situaciones planteadas por los robots autónomos. En muchas ocasiones se utilizan robots que cuentan con un mando remoto (robots teleoperados), aunque los principales centros de investigación en Robótica han orientado desde hace tiempo sus esfuerzos hacia los sistemas robóticos autónomos.

1.1. Estructura de Tareas en Robótica

A un ser humano, aparentemente, la capacidad de planificar sus propias acciones no le plantea una gran dificultad puesto que, en buena parte, la interacción con su entorno se realiza de forma inconsciente. Esta inconsciencia se torna en desconocimiento a la hora de intentar duplicarla en un programa de ordenador que se encargue de controlar un robot. Para llevar a cabo la emulación de actividades ([?]) realizadas por el ser humano se necesita desarrollar tecnologías como el razonamiento automatizado, la percepción y el control. Para dar una idea de la enorme complejidad del tema, en los siguientes apartados se van a comentar las disciplinas y las tareas implicadas más directamente en el desarrollo de robots autónomos, tomando como base el diagrama que aparece en la figura 1.3.

Como el robot realiza una tarea ejecutando secuencias de movimientos en un espacio de trabajo ocupado por objetos, es necesario, por un lado, estudiar su cinemática y por otro, disponer de una descripción del entorno, lo que constituye el objetivo de los dos apartados siguientes.

1.1.1. Cinemática

La cinemática se interesa por la descripción analítica del movimiento espacial del robot, con respecto a un sistema de referencia, como una función del tiempo. En cualquier instante de tiempo es necesario determinar el conjunto de parámetros (configuración) que definen la posición y orientación de todos los puntos del robot con respecto de un sistema de referencia fijo.

La determinación de este conjunto de parámetros está ligado íntimamente con la geometría del robot, ([?]) pudiéndose diferenciar entre robots formados por un único elemento rígido y los que cuentan con una serie de elementos, rígidos o no, unidos por articulaciones. Asimismo, existen robots que se pueden mover libremente dentro de un espacio de trabajo mientras que otros tienen un punto fijo.

Dependiendo de su geometría el robot podrá adoptar una serie de configuraciones, denominándose espacio de las configuraciones del robot (C-espacio) al conjunto de todas sus posibles configuraciones. Este concepto que es esencialmente una herramienta de representación, fue introducido en Robótica por Lozano-Pérez, y ocupará un papel fundamental en este trabajo de investigación. Cada punto del C-espacio es una tupla de una cierta dimensión, donde se especifican los valores para los parámetros que corresponden con los grados de libertad del robot. La posición y la orientación del robot en su espacio de trabajo queda completamente determinada si se especifican los valores para los grados de libertad del robot.

Por ejemplo, supóngase que el robot es un disco que puede moverse y girar libremente en el plano con unas determinadas dimensiones $[0, a] \times [0, b]$, tal y como aparece en la figura 1.1. Para describir una configuración arbitraria del robot en un determinado punto del plano hay que especificar las coordenadas de ese punto, con respecto a un sistema de referencia fijo. Por tanto, una configuración vendría dada por la tupla de dimensión dos (x, y) . Su espacio de las configuraciones estaría formado por todos los pares de coordenadas (x, y) , donde x e y son números reales que pueden tomar valores en un intervalo cuyas dimensiones dependen de las del espacio de trabajo.

En la figura 1.1 aparece un robot articulado en el plano, cuya cadena cinemática está formado dos elementos conectados mediante una articulación de revolución. Además, el primer elemento puede girar respecto a un eje perpendicular al plano. Una configuración arbitraria del robot viene especificada dando el valor del ángulo que forma el primer elemento con el eje x y el valor del ángulo que forma el segundo elemento respecto del primero. Por tanto, una configuración viene dada por la tupla de bidimensional (θ_1, θ_2) . En este caso el espacio de las configuraciones está formado por todas las tuplas (θ_1, θ_2) con $\theta_1, \theta_2 \in [0, 2\pi)$.

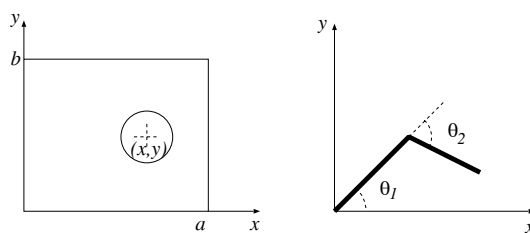


Figura 1.1: Robot móvil y robot articulado en el plano

1.1.2. Percepción

Los manipuladores más implantados en la industria desarrollan su trabajo en un entorno perfectamente estructurado, donde se conocen de antemano las posiciones donde se encuentran los objetos que tienen que manejar o que forman parte del mobiliario industrial. Sin embargo, los robots autónomos necesitan disponer de información del entorno con el cual deben interactuar. Por ejemplo, si se trata de un robot móvil encargado de transportar objetos por un pasillo es necesario que disponga de la situación de los elementos de su entorno, que lógicamente se modifica.

En este sentido, la capacidad sensorial del robot se encarga de realizar las labores de percepción de información y juega un papel fundamental en el desarrollo de los sistemas robóticos inteligentes. En principio, hay una variedad de sensores que son utilizados en Robótica ([?]). Pero los de visión ocupan un papel fundamental. A partir de la información recogida por ellos se puede llegar a disponer de una descripción del entorno, en concreto de todos los puntos del espacio de trabajo que están ocupados por los objetos. En este trabajo de investigación se supondrá que se dispone ya de una información exacta y completa de su entorno.

Realizar en el espacio de las configuraciones algunas de las tareas implicadas en el desarrollo de robots autónomos presenta grandes ventajas, como se analizará en apartados sucesivos. Para beneficiarse de estas ventajas es necesario representar a los obstáculos en el espacio de las configuraciones. La descripción geométrica del robot junto con la descripción de su entorno permitirán conocer el conjunto de configuraciones del robot que producen colisión con los obstáculos

(figura 1.3).

En este espacio también deben estar representadas las restricciones que el propio robot presenta sobre su movimiento. Entre éstas destacan las debidas a las colisiones con los distintos elementos que componen la cadena cinemática del robot, como ocurre en los articulados. Otro conjunto de restricciones son las que impone el propio diseño de la cadena cinemática, por ejemplo los topes mecánicos en las articulaciones.

Con estas consideraciones, los puntos del espacio de las configuraciones se pueden clasificar en dos grupos: prohibidos y permitidos. El primero está constituido por las configuraciones donde el robot puede chocar con los obstáculos, chocar con alguno de sus elementos y sobrepasar alguna restricción mecánica. El segundo grupo de puntos lo componen aquellas configuraciones que son seguras para el robot y libres de algún tipo de colisión. El conjunto de configuraciones prohibidas se denomina *obstáculo en el espacio de las configuraciones* o *C-obstáculo*. La determinación de forma explícita de este conjunto de configuraciones prohibidas constituye el objetivo en el cual se ha centrado este trabajo de investigación.

1.1.3. Planificación de tareas

Como se puede observar en la figura 1.3, la planificación de tareas se encuentra en el nivel superior de esta jerarquía de disciplinas implicadas en la creación de robots autónomos. Un planificador de tareas se puede ver como un bloque lógico con una entrada y una salida. La entrada sería una descripción de la tarea que el usuario desea que ejecute el robot y la salida sería el conjunto de operaciones necesarias para su realización ([?]).

Por ejemplo, supóngase que el robot es un manipulador que se encuentra en una planta de fabricación de bolígrafos, más concretamente en la línea de montaje. La entrada al planificador sería *montar un bolígrafo*. El planificador dividiría la tarea en las cuatro operaciones siguientes:

1. coger la carcasa y colocarla en un soporte,
2. coger la vaina con la recarga de tinta e introducirla dentro de la carcasa,
3. coger el capuchón y situarlo en un extremo,

4. transportar el bolígrafo a la línea de embalado.

A pesar de la sencillez del ejemplo, en general, la etapa de planificación de tareas puede ser sumamente compleja, porque la tarea en sí misma lo sea o porque se deseen introducir mejoras adicionales, como pueden ser la tolerancia a fallos, la optimización del tiempo de ejecución, la integración con el resto de los componentes de línea de fabricación, etc.

1.1.4. Planificación de movimientos

Se ha mostrado como el planificador de tareas ha desglosado una tarea compleja en un conjunto de operaciones más sencillas, donde cada una de éstas consiste en coger una pieza en un determinado punto del espacio y colocarla en otro. Las posiciones iniciales y finales de cada subtarea constituyen las entradas al planificador de movimientos (figura 1.3). Este se encarga de encontrar un camino que conecte el punto origen y el punto destino sin que se produzcan colisiones con los obstáculos que ocupan parcialmente su entorno de trabajo, o devolver un fallo si este camino no existe.

Por ejemplo, para la tercera operación del montaje de un bolígrafo, las entradas al planificador de movimientos serían la posición A donde se encuentra el capuchón y la posición B del extremo del bolígrafo. Su salida sería la secuencia de movimientos que le permiten desplazarse desde A a B, sin que choque con los objetos que se encuentran en su entorno de trabajo o viole alguna de las restricciones que el robot mismo impone sobre su movimiento.

Un punto de vista incorrecto y muy extendido es el que considera que la planificación de movimientos consiste esencialmente en detectar colisiones. Además, se ocupa de calcular caminos libres de colisiones entre obstáculos móviles, de coordinar el movimiento de varios robots, de planificar movimientos para empujar y deslizar objetos con el fin de lograr relaciones exactas entre estos, de planificar la manera de coger los objetos de forma estable, etc.

1.1.5. Control de movimientos

El resultado obtenido por el planificador de movimientos es un camino que especifica la secuencia continua de configuraciones que el robot debe recorrer

para alcanzar la configuración final. La tarea fundamental del controlador en tiempo real es lograr que el robot realice esta secuencia de movimientos, es decir, siga el camino generado, actuando sobre él. En concreto, dado un camino, el controlador tiene que encontrar la función temporal que define los pares a aplicar a los actuadores del robot en cada instante de tiempo y aplicarlos ([?]).

Normalmente esta etapa se puede dividir en dos pasos (figura 1.3). El primero, denominado generación de trayectorias, consiste en definir el perfil de velocidad a lo largo del camino. Este paso se puede realizar antes de la ejecución del movimiento. El segundo, denominado seguimiento de la trayectoria, consiste en calcular los pares que se deben aplicar a los actuadores en cada momento para realizar el movimiento deseado. En este paso se utiliza, directa o indirectamente, la dinámica del robot para calcular el par que tiene que ser aplicado a cada actuador. Si la dinámica utilizada por el controlador fuera un modelo perfecto, no sería necesario disponer de un sistema realimentado. Sin embargo, debido a las diferentes perturbaciones y a la inestabilidad, es necesario disponer de sensores para determinar la desviación entre el estado deseado y el estado actual del robot. Mientras se está ejecutando el movimiento, el controlador calcula los pares que tienden a eliminar esta desviación.

La figura 1.2 muestra la relación entre el planificador de movimientos, el generador de trayectorias, el controlador y el robot. El camino τ proporcionado por el planificador de movimientos es la entrada al generador de trayectorias, que determina la dependencia temporal de los parámetros que definen una configuración. La salida del generador de trayectorias, que son las configuraciones deseadas q_d como una función del tiempo, se introduce en el controlador. Este calcula los pares P que han de ser aplicados a cada actuador, a partir de la diferencia entre la configuración actual q_s , medida por los sensores, y la configuración deseada q_d .

La arquitectura presentada en la figura 1.2 es clásica, ya que separa la etapa de planificación de movimientos de la etapa de control. Su ventaja al dividir el problema es decidir un movimiento en varias etapas claramente definidas. Pero esto podría llevar a movimientos poco eficientes. Por ejemplo, si el controlador intenta minimizar el tiempo de ejecución, la geometría de los caminos previamente calculada puede que no permita al controlador aprovechar completamente

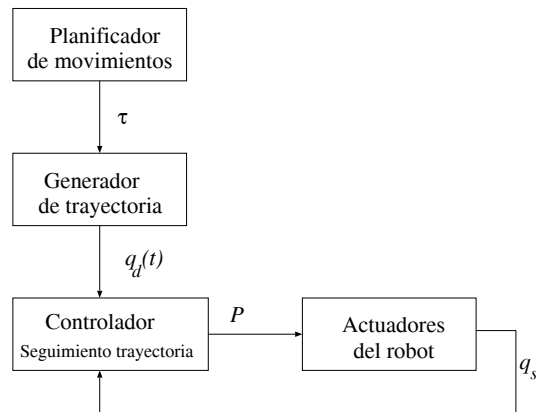


Figura 1.2: Controlador de movimientos

la capacidad de los actuadores. Probablemente, un mejor planteamiento es tener en cuenta, durante la planificación de movimientos, la dinámica del robot junto con los límites de saturación de los actuadores y, así, generar caminos que permitan ejecutar movimientos cuasi-óptimos en el tiempo. Se han propuesto varias técnicas para implementar este planteamiento [?], [?], [?], [?]. Estas técnicas podrían llegar a ser particularmente importantes cuando el robot trabaje en un entorno dinámico entre obstáculos móviles, cuando la tarea se tenga que efectuar dentro de unos estrictos límites de tiempo (por ejemplo, debido a que se relacione con un proceso externo), o cuando se tenga que optimizar la productividad del robot (por ejemplo, en fabricación).

1.2. Ventajas de Trabajar en el Espacio de las Configuraciones

En esta sección se van a analizar las principales ventajas que se obtienen al realizar las tareas implicadas en el desarrollo de robots autónomos en espacio de las configuraciones en lugar de en el espacio de trabajo, tomando como base la descripción de tareas de la figura 1.3.

Como característica fundamental hay que resaltar el hecho de que en el espacio de las configuraciones el robot se reduce a un punto. A lo largo de este apartado, se expondrá cómo se pueden beneficiar de este hecho la planificación

de movimientos, la generación de trayectorias y el control de movimientos. Sin embargo, aparece el problema adicional de obtener el conjunto de configuraciones prohibidas (C-obstáculos), que engloba a las configuraciones que producen colisión, bien con los obstáculos o con alguno de los elementos de su cadena cinemática, y las que sobrepasan alguna restricción mecánica. Así, también se expondrá cómo se puede optimizar esas tres tareas, si se dispone del conjunto de restricciones, tanto internas como externas, sobre el movimiento del robot.

La planificación de movimientos se puede realizar tanto en el espacio de trabajo como en el espacio de las configuraciones. Trabajando en el espacio de las configuraciones se reduce drásticamente el problema de encontrar la secuencia de configuraciones que el robot tiene que recorrer hasta llegar a uno de los destinos intermedios, ya que supone planificar los movimientos de un punto. Sin embargo, realizar esta tarea en el espacio de trabajo supone planificar los movimientos de los distintos elementos que forman parte del robot.

Además, tiene como ventaja adicional que la comprobación de si existe o no colisión del robot con los obstáculos o entre sus elementos se reduce a la verificación de si el punto que representa al robot pertenece o no a los obstáculos representados en dicho espacio, es decir, si pertenece o no al C-obstáculo.

Los algoritmos de seguimiento de trayectorias se pueden simplificar si se trabaja en el espacio de las configuraciones, ya que las variables que definen una configuración constituyen las referencias para el controlador. Además, dichos algoritmos se pueden beneficiar de disponer explícitamente de las restricciones que los obstáculos imponen en las variables de control.

Asimismo para los algoritmos de generación de la trayectoria es muy útil conocer las configuraciones que producen colisión o no con los obstáculos. De esta forma el perfil de velocidad se puede generar teniendo en cuenta la distancia entre la configuración planificada y la que produce colisión, entre otros criterios.

En el planteamiento expuesto previamente, donde se consideraban de forma conjunta las etapas de planificación y control de movimientos, se tuvieron en cuenta las restricciones en el control debidas a la naturaleza de los actuadores. Sin embargo, no se consideraron aquellas que son producidas por la presencia de obstáculos en el espacio de trabajo. La mayoría de los trabajos que abordan la planificación dinámica no disponen de una representación de los obstáculos

en un espacio de las configuraciones. Esto supone una dificultad adicional a las que esta filosofía presenta, pues hay que considerar que se trata de un problema de optimización temporal de elevada complejidad. El disponer explícitamente de una representación de los obstáculos en el espacio de las configuraciones no resuelve el problema, pero simplifica de forma muy significativa su consecución.

1.3. Antecedentes

En este apartado se revisarán los principales trabajos de investigación que se han realizado sobre la representación de los obstáculos en el espacio de las configuraciones. De esta forma se analizará el estado actual de este área de investigación, con el objetivo de poner de manifiesto las principales aportaciones de este trabajo.

Como ya se ha comentado, la resolución de un problema de planificación o de control resulta mucho más fácil en el espacio de las configuraciones, C-espacio, que en el espacio de trabajo del robot, ya que en el primero de ellos la posición y la orientación del robot se caracterizan con un único punto. En él cada coordenada representa un grado de libertad en la posición o en la orientación del robot. Esta idea de reducir el robot a un punto en un espacio apropiado fue introducida por Udupa [?], aunque el término *espacio de las configuraciones* todavía no se había utilizado. Más tarde, Lozano-Pérez [?] adoptó la noción de *espacio de las configuraciones* de la Mecánica y la popularizó en la planificación de movimientos.

Como se podrá observar a lo largo de esta revisión, el avance en el cálculo del C-espacio siempre ha ido ligado a la planificación de movimientos. Sin embargo, también la noción de espacio de las configuraciones se ha utilizado directamente en control [?] o incluso intentando unificar ambas etapas [?].

En la planificación de caminos, existen métodos que no necesitan representar de forma explícita los obstáculos en el C-espacio. Entre ellos, un primer grupo realizan la detección de colisiones únicamente para los caminos calculados, como en [?], [?] y [?]. Un segundo grupo [?] supone que existe el C-espacio y el movimiento se planifica sobre o entre superficies del C-espacio, aunque éstas no se calculan de forma explícita. Los algoritmos de planificación que se encuentran

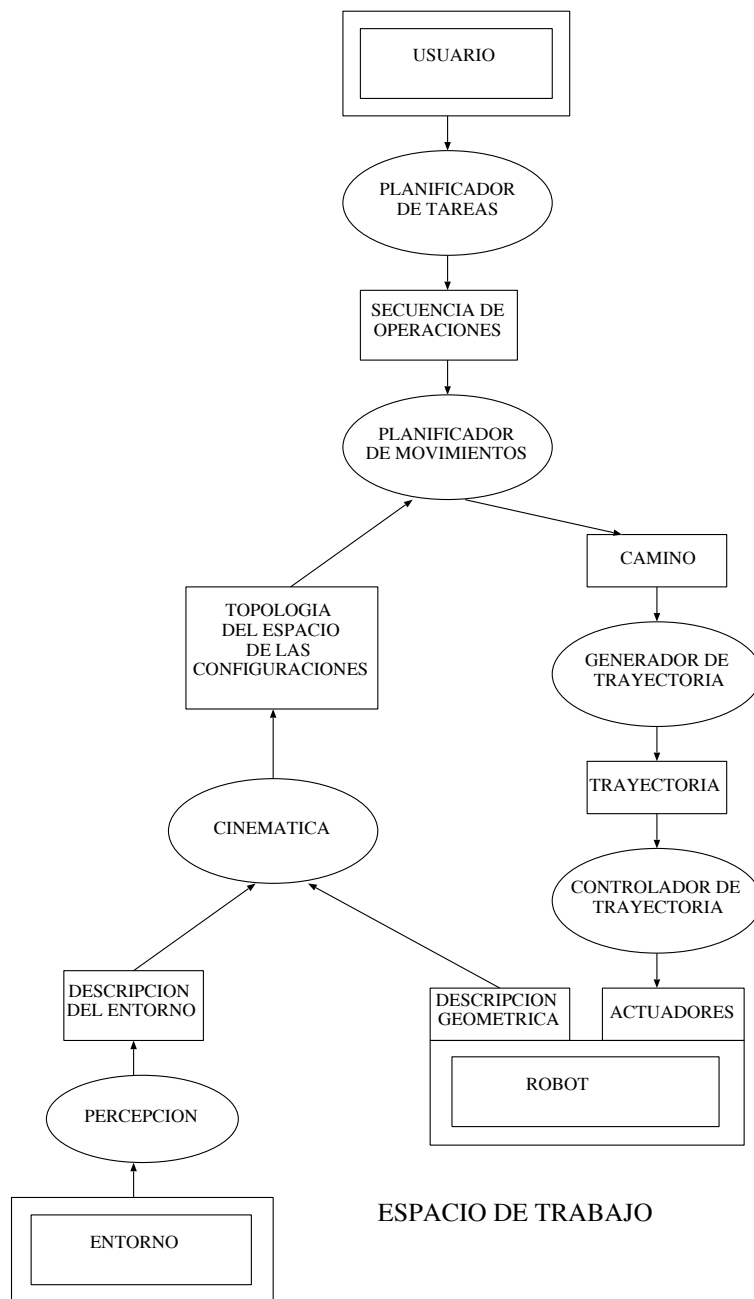


Figura 1.3: Diagrama de tareas en Robótica

en uno de ambos grupos pierden información global sobre el espacio de las configuraciones, lo que puede conllevar a problemas en la planificación, tales como oscilaciones, caídas en mínimos locales, etc.

Entre los algoritmos de planificación que calculan los C-obstáculos explícitamente ([?] [?] [?] [?] [?] [?]) se encuentran los que abordan el problema en una única etapa, mientras que otros lo realizan en dos etapas separadas. Este último planteamiento fue propuesto por Lozano-Pérez [?], que divide la tarea de planificación de caminos en:

1. el problema de calcular el conjunto de configuraciones seguras, denominado *findspace*,
2. el problema de calcular una secuencia de configuraciones a partir del conjunto de configuraciones seguras, denominado *findpath*.

La representación de los obstáculos en el espacio de las configuraciones (C-obstáculos), aunque es muy útil como punto de partida en el problema de *findpath*, está relacionada exclusivamente con el problema de *findspace*.

Dentro de los trabajos que siguen el planteamiento propuesto por [?] se encuentra un grupo ([?] [?] [?] [?]) donde en la etapa de *findspace* se genera, además, una descripción topológica sobre los C-obstáculos. Esta se puede utilizar en la etapa de *findpath* como información adicional para guiar la búsqueda.

De todas formas, tanto los que calculan directamente los obstáculos en el C-espacio como los que los utilizan de forma implícita, resaltan el problema del tiempo de cálculo que esta representación lleva asociado. En general, como se mostrará seguidamente, depende del número de vértices del robot y de los obstáculos.

La mayoría de los trabajos que calculan explícitamente los C-obstáculos tratan el problema de un objeto rígido que se mueve libremente (robot móvil) en un espacio de trabajo ocupado parcialmente por obstáculos. Normalmente, consideran que los objetos en el espacio de trabajo, tanto obstáculos como el robot, están representados por modelos algebraicos. Esto incluye que ambos se puedan representar mediante polígonos o poliedros. Inicialmente se restringen a objetos convexos y, posteriormente, contemplan objetos no convexos. Esta extensión supone una carga computacional adicional, asociada con la descomposición en objetos convexos.

En [?] se propone un método para calcular la representación del límite de un C-obstáculo cuando los objetos son polígonos convexos. El algoritmo propues-

to es bastante óptimo, pues calcula los vértices del C-obstáculo, que también es convexo, en un tiempo de cálculo c , siendo n_A y n_B el número de vértices del robot y del obstáculo, respectivamente. En [?] se plantea un algoritmo con un tiempo de cálculo de $O(n_A^2 n_B^2 \log(n_A n_B))$ cuando el robot (que se traslada y no rota) y los obstáculos son polígonos no convexos. En [?] y en [?] se describen métodos generales para polígonos convexos y no convexos, cuando el robot se desplaza y gira libremente, y se obtiene el límite del C-obstáculo en un tiempo $O(n_A^3 n_B^3)$ y $O(n_A^3 n_B^3 \log(n_A n_B))$, respectivamente. En cuanto a espacios de trabajo poliédricos, en [?] se presenta una extensión de su algoritmo poligonal; en [?] se calcula la representación del límite del C-obstáculo en $O(n_A + n_B + n_A n_B)$; y el algoritmo propuesto en [?] construye el C-obstáculo en $O(n_A^3 n_B^3 \log(n_A n_B))$.

Todos estos procedimientos proporcionan una representación explícita del límite de un C-obstáculo para un robot móvil, como una lista de caras, lados y vértices, con sus ecuaciones y su relación topológica de adyacencia. Existen otros trabajos ([?], [?], [?] y [?]) que siguen otra línea de investigación claramente diferente y conceptualmente más sencilla. En concreto, proponen utilizar la lógica como herramienta matemática y, así, representar el C-obstáculo para un robot móvil mediante un predicado. En estos trabajos se presentan distintas expresiones para el predicado, considerando, inicialmente, espacios de trabajo poligonales con objetos convexos y, posteriormente, su extensión a espacio poliédricos.

Como se puede observar, el tiempo de cálculo de todos los algoritmos que construyen los obstáculos en el espacio de las configuraciones depende directamente del número de vértices del robot móvil y de los obstáculos.

Mientras que existe un amplio grupo de trabajos centrado en robots móviles, sólo un grupo reducido ([?] [?] [?] [?]) se ocupa del estudio de manipuladores articulados en espacios de 2 ó 3 dimensiones. Entre ellos existen dos tendencias claramente separadas: una donde se trata a los robots articulados como un conjunto de móviles al que se aplican técnicas que provienen de éstos últimos; y otra, donde se consideran como una unidad y se plantean nuevas técnicas.

Dentro de los que siguen la primera tendencia ([?] [?] [?]) se encuentran los que aplican, con ciertas modificaciones, la idea de la descomposición aproximada

en celdas, propuesta para móviles. El método concreto consiste en discretizar el intervalo de movimiento de cada articulación en intervalos más pequeños y, después, comprobar si el robot colisiona o no con los obstáculos cuando las distintas articulaciones se encuentran en estos subintervalos. Este proceso lleva asociada una gran carga computacional, pues habría que realizar el test de colisiones para todos los elementos en todas las configuraciones consideradas.

En cuanto a los que siguen la segunda tendencia ([?] [?] [?] [?]) intentan calcular las ecuaciones exactas del límite de los obstáculos en el espacio de las configuraciones. Consideran, por un lado, las ecuaciones algebraicas que definen los obstáculos y, por otro, las expresiones matemáticas que describen la cinemática del robot. A partir de las condiciones que se establecen cuando los elementos del robot intersectan con los obstáculos se calculan las ecuaciones explícitas del límite de los C-obstáculos.

Uno de los primeros análisis sobre la geometría de los C-obstáculos para distintos brazos planares fue presentado en [?], donde se pone de manifiesto su fuerte dependencia con la cinemática inversa del manipulador. Existen trabajos ([?] [?] [?] [?]) en los que se calculan, de forma particular, las ecuaciones de los C-obstáculos para manipuladores planares de revolución y se analiza su dependencia con distintos parámetros. Posteriormente, proponen extender estos algoritmos para robots tipo PUMA, utilizando la técnica de *slicing*. Un método más general para obtener las ecuaciones del límite del C-obstáculo para manipuladores planares con n elementos se propone en [?]. Se aplica a un robot planar 2R y a dos robots planares 2R cooperantes ([?]). Después, en [?], se generaliza para robots espaciales y obstáculos poliédricos.

Todos los métodos que siguen la segunda tendencia simplifican la geometría de los elementos del manipulador y de los obstáculos. Esto conduce a no disponer de una representación exacta de los obstáculos en el espacio de las configuraciones. Además, no todas las técnicas examinan las colisiones con todos los elementos del manipulador, sino que sólo consideran los choques con el efector final.

En [?] se propone obtener la representación de los obstáculos en el espacio de las configuraciones mediante una convolución algebraica de un obstáculo y un robot móvil. Esta idea se encuentra recogida implícitamente en [?] donde,

para un robot A móvil, se define la proyección de un obstáculo B del espacio de trabajo en un obstáculo $CO_A(B)$ del espacio de las configuraciones como $CO_A(B) \equiv \{x \in \text{Cspace}_A / (A)_x \cap B \neq \emptyset\}$. En [?] se plantea explícitamente que esta operación se puede ver como la convolución del conjunto A con el conjunto B . Esta idea es la base de nuestro trabajo, donde se propone un formalismo matemático para calcular los C-obstáculos utilizando la convolución de dos funciones. En general, este formalismo se puede establecer para robots móviles y articulados, tanto en entornos de trabajo poligonales como poliédricos.

En los algoritmos de planificación propuestos en los últimos años se ha planteado trabajar en espacios de las configuraciones discretos [?]. Concretamente, se utiliza una matriz binaria para representar los obstáculos en el C-espacio, donde una determinada celda asociada a unos determinados valores discretos de los grados de libertad del robot puede tomar dos valores: '1' si es una configuración prohibida y '0' si pertenece al espacio libre. De esta forma se consigue reducir de forma drástica el tiempo de cálculo que la etapa de planificación dedica a la detección de colisiones. La obtención de la matriz binaria para representar el C-espacio se puede abordar de dos formas: discretizando el espacio de las configuraciones calculado de forma analítica [?] o calculando directamente una representación discreta de los C-obstáculos. En [?] se transforma, de forma analítica, un conjunto de primitivas desde el espacio de trabajo al espacio de las configuraciones y el resultado se almacena en matrices de bits. Basándose en las propiedades de los obstáculos en el C-espacio, las transformaciones de estas primitivas se combinan para calcular matrices de bits que representan a objetos más complejos en el C-espacio. Una idea similar aparece en [?] donde, partiendo de una matriz de bits que representa a los obstáculos en el espacio de trabajo y de los bitmaps de un conjunto de primitivas en el C-espacio, se calcula una matriz de bits de los obstáculos en el C-espacio. Ambos trabajos se ocupan de manipuladores articulados: el primero con dos y tres grados de libertad y el segundo llega a trabajar con seis grados de libertad.

Kavraki [?] propone un algoritmo para calcular una matriz binaria que representa al C-espacio de un móvil bidimensional como la convolución de una matriz binaria de los obstáculos en el espacio de trabajo y de otra del robot. Para realizar la convolución eficientemente utiliza la Transformada Rápida de

Fourier. Con el método propuesto en [?] se obtiene un bitmap del espacio de las configuraciones más exacto que con los que utilizan primitivas elementales. Además es independiente del número de obstáculos, de la forma de estos y de la del robot y, únicamente, depende de la resolución con la que se discretiza. En dicho trabajo solamente se tiene en consideración una plataforma bidimensional que se mueve libremente sobre un plano. Para esta situación particular la idea de aplicar la convolución se puede intuir con facilidad, y no así para articulados. Sin embargo, no aporta ningún formalismo matemático que permita concluir esa idea y extender el planteamiento a otras situaciones con mayor complejidad, como los robots articulados en 3 dimensiones.

1.4. Objetivos de este Trabajo

En los apartados previos ha quedado justificada la necesidad de representar los obstáculos en el espacio de las configuraciones para simplificar las tareas que intervienen en el desarrollo de robots autónomos. Sin embargo, la generación de esta representación conlleva una elevada carga computacional asociada. Además, no existen métodos rápidos que traten el problema de forma general, desde el punto de vista del robot y de los obstáculos.

Tomando esta realidad como base, el objetivo fundamental de este trabajo es proponer un nuevo método general que permita obtener una representación de los obstáculos en el espacio de las configuraciones de un robot, reduciendo dicha carga computacional, y pueda emplearse en tareas de planificación y control.

Este método debe estar sustentado, de forma teórica, por un formalismo matemático sólido y de carácter general. El punto clave de este formalismo será la proposición de una nueva expresión matemática para calcular los C -obstáculos que sea equivalente a otras propuestas en la bibliografía.

La aplicabilidad del método debe abarcar tanto a robots móviles como articulados, en dos y tres dimensiones, sin necesidad de realizar ninguna adaptación. Un análisis detallado de los antecedentes bibliográficos en la materia permite poner de manifiesto las diferencias entre las técnicas utilizadas hasta el momento para ambas estructuras y la carencia casi absoluta de resultados para los manipuladores ampliamente utilizados en espacios tridimensionales. Además, la

mayor parte de los trabajos que existen se restringen al estudio de los contactos con el efector final, y no consideran los choques con todos los elementos. La concreción del método a manipuladores articulados constituirá uno de los puntos centrales de este trabajo. Esto permitirá obtener una representación explícita de los obstáculos en el C-espacio para manipuladores como el PUMA, el SCARA y el Stanford, y examinar los choques con todos los elementos y no sólo con el efector final.

El método debe de ser aplicable a objetos reales, sin restringirse a un conjunto limitado de objetos y sin perder exactitud. En la mayoría de las técnicas existentes bien se realizan simplificaciones en la definición de la geometría de los objetos o bien se consideran unas formas geométricas determinadas. Esto es debido a que la complejidad computacional de dichas técnicas depende de forma drástica de diversas características geométricas de los objetos, como son el número de vértices y caras, de si son cóncavos o convexos, etc. Como consecuencia, se debía tomar una decisión de compromiso entre la alta carga computacional y la pérdida de exactitud.

El diseño de un método de carácter general permitirá, por tanto, resolver de forma precisa cualquier situación y su aplicación tendrá como resultado algoritmos que puedan ser implementados de forma sencilla. Para cada robot particular, el argumento de entrada deberá ser una estructura donde se representen fácilmente los obstáculos. Así, si se dispone de la capacidad sensorial adecuada, los algoritmos podrán ser aplicables a entornos que varíen de forma dinámica.

La complejidad computacional de los algoritmos debe de ser independiente de la geometría tanto de los objetos que componen el robot como de los situados en su entorno, sin ninguna contraprestación en cuanto a la exactitud de la representación obtenida.

El resultado final de estos algoritmos será una estructura que contenga el conjunto de configuraciones prohibidas. Ésta tendrá que poder utilizarse con facilidad en la etapa de planificación de caminos o en la de seguimiento de la trayectoria. Asimismo, la estructura generada deberá resultar independiente del tipo de robot y, en definitiva, transparente para el planificador y para el controlador.

Bajo el carácter general del método subyace una interpretación física para el formalismo propuesto. Encontrar esa interpretación y analizarla constituirá el objetivo final de este trabajo de investigación.

1.5. Estructura de la Memoria

El resto de la memoria se organiza de la siguiente forma.

En el capítulo 2 se definen, de forma rigurosa, los conceptos básicos ya introducidos que se van a utilizar a lo largo de este trabajo de investigación.

En la sección 1.3 de esta introducción se han revisado los antecedentes más destacados relacionados con este área de investigación. Una revisión mucho más detallada se expone en el capítulo 3, donde se analizan por separado las técnicas que han propuesto diferentes investigadores para robots móviles y para manipuladores articulados. Con ella se mostrarán las ventajas, inconvenientes y limitaciones de las técnicas existentes. Esto permitirá valorar las ventajas del formalismo matemático propuesto y del método que de él se deduce, los cuales se presentan en el capítulo siguiente.

Así, en el capítulo 4 se propone un formalismo matemático que permite calcular la representación de los obstáculos en el espacio de las configuraciones, con la inclusión de una nueva expresión matemática para realizar el cálculo analítico. En esta expresión se define una función que permitirá conocer si el robot en una configuración dada colisiona o no con los obstáculos. Además, se incluye una demostración de que esta expresión es equivalente a otra ampliamente difundida. Posteriormente, se muestra el procedimiento que permite simplificar la evaluación de dicha función, a través de una correcta elección de los sistemas de referencia, así como de las coordenadas de trabajo.

En los capítulos siguientes, 5 y 6, se aplica el método propuesto a los principales tipos de estructuras robóticas: robots móviles y manipuladores articulados. Así, en el capítulo 5 se presentan las expresiones obtenidas para los diferentes casos: un objeto bidimensional móvil con capacidad de traslación, de traslación-rotación y, finalmente, un objeto tridimensional móvil. Como ya se ha comentado, alguno de estos robots está bastante estudiado en la literatura y se incluye con el objetivo de validar la metodología propuesta en este trabajo

de investigación.

Como se señalaba en la sección de objetivos, los trabajos existentes se ocupan en su mayor parte de los robots móviles, mientras que para los manipuladores articulados se aplican técnicas heredadas de los anteriores con resultados poco alentadores. Por esta razón, la aplicación de la técnica propuesta a este tipo de robots es un objetivo fundamental que se abordará en el capítulo 6. Inicialmente, se va a analizar un caso ampliamente estudiado, como es el manipulador planar de revolución. Posteriormente, se aplicará el formalismo para los manipuladores espaciales más extendidos en el ámbito industrial: PUMA, SCARA y el de Stanford.

Puesto que el objetivo final es el desarrollo de algoritmos para el cálculo de la representación de los obstáculos en el espacio de las configuraciones, en el capítulo 7 se realiza, en primer lugar, una discusión sobre el proceso de discretización de las expresiones obtenidas en los apartados anteriores. A continuación se presentan los diferentes algoritmos que se obtienen para cada una de las estructuras robóticas consideradas en los dos capítulos anteriores. Para algunas situaciones concretas se muestran los resultados que se obtienen con los algoritmos propuestos. Este capítulo se puede considerar como la aportación más importante de resultados de este trabajo.

En el capítulo 8 se presenta un resultado adicional de la formulación teórica planteada, dando una nueva interpretación para su expresión matemática central. Tomando como caso particular el robot circular, se podrá comprobar que esta formulación permite obtener las superficies de potencial artificial que se generan en el espacio de las configuraciones cuando se considera que los obstáculos producen un potencial repulsivo. Este aspecto se presenta como un punto partida para futuros desarrollos de investigación de tesis doctoral.

Finalmente, en el capítulo 9 se analizan las principales conclusiones de este trabajo.

Capítulo 2

Obstáculos en el Espacio de las Configuraciones

Como se ha comentado previamente, la idea fundamental del espacio de las configuraciones es representar el robot como un punto en un espacio apropiado - el espacio de las configuraciones del robot - y proyectar los obstáculos en este espacio. Esta proyección simplifica de forma drástica las tareas de planificar y controlar los movimientos de un robot. Por un lado, el problema de planificación se reduce a planificar los movimientos de un punto y, además, se tienen de forma explícita las configuraciones que producen colisiones. Por otro lado, es bastante común identificar un punto del espacio de las configuraciones con la referencia de los controladores que se ocupan de gobernar los movimientos del robot. En este contexto la proyección de los obstáculos en este espacio se puede considerar como restricciones en el problema de control.

Esta proyección depende tanto de los obstáculos como del tipo de robot considerado. En cuanto a los primeros, no se impondrá ninguna limitación debida a su forma geométrica, número de vértices, etc. Respecto al segundo, una clasificación muy genérica es aquella que diferencia entre robots que pueden moverse libremente en su espacio de trabajo, denominados móviles, y aquellos que tienen un punto fijo. Otra los clasifica en robots articulados o no-articulados, dependiendo de si su estructura mecánica está formada por uno o varios elementos que se mueven. Tomando como base estos criterios de clasificación existirían cuatro

tipos de estructuras robóticas. Los robots más implantados en las aplicaciones más usuales, y que son tratados en este trabajo de investigación, son los móviles constituidos por un único elemento y los articulados con un punto fijo. Existen aplicaciones muy específicas donde se utilizan robots móviles articulados, aunque este tipo no será tenido en cuenta en este trabajo.

En este capítulo se va a presentar un conjunto de nociones necesarias para definir el concepto de espacio de las configuraciones de un robot y la representación de los obstáculos en este espacio, para un objeto rígido que se mueve. De forma directa e inmediata estas definiciones generales son aplicables a objetos rígidos que se mueven libremente. Posteriormente, se van a extender dichas definiciones para robots articulados con una base fija. Se presentarán, de forma conjunta con ellas, algunos ejemplos sobre robots móviles y manipuladores articulados con distinta forma geométrica que ayudarán a comprender estos conceptos.

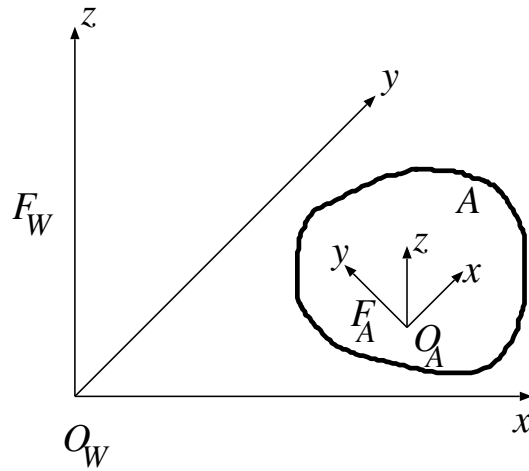
2.1. Espacio de las Configuraciones de un Objeto Rígido Móvil

Aunque los conceptos de configuración y espacio de las configuraciones ya fueron introducidos en el capítulo previo, en esta sección se presentan de forma más rigurosa, incluyendo unos casos particulares que ayudarán a comprenderlos.

Sea \mathbf{A} un objeto rígido - el robot - que se mueve en un espacio de trabajo W (figura 2.1). Se representa W como un espacio Euclídeo R^n de dimensión n , donde $n = 2$ ó 3 . En él se define un sistema de referencia fijo denominado F_W . Se representa \mathbf{A} en una posición y orientación de referencia como un subconjunto compacto de R^n . En \mathbf{A} se establece un sistema de referencia, F_A , que se mueve con él, de tal forma que cada punto en el robot tiene unas coordenadas fijas respecto a F_A . Los orígenes de ambos sistemas F_W y F_A se denotan por O_W y O_A , respectivamente. O_A se denomina el punto de referencia de A .

Definición 1 *Una configuración q de \mathbf{A} es una especificación de la posición y la orientación de F_A con respecto a F_W .*

La configuración de referencia de \mathbf{A} , que se denota por 0 , es una única configuración de C seleccionada de forma arbitraria.

Figura 2.1: Robot móvil en $W = R^3$

Definición 2 *El espacio de las configuraciones de \mathbf{A} es el espacio C de todas las posibles configuraciones de \mathbf{A} .*

El espacio de las configuraciones es intrínsecamente independiente de la elección de los sistemas de referencia F_A y F_W . Solamente la representación de C depende de la elección de estos sistemas.

El subconjunto de W ocupado por \mathbf{A} en una configuración q se denota por $\mathbf{A}_{(q)}$ o $\mathbf{A}(q)$. El robot en su configuración de referencia se denota por $\mathbf{A}_{(0)}$ o $\mathbf{A}(0)$. Cuando \mathbf{A} está en la configuración q , un punto a de \mathbf{A} se denota por $a(q)$ en W . Así, para dos configuraciones q y q' cualesquiera, $a(q)$ y $a(q')$ son el mismo punto de \mathbf{A} , pero en general no coinciden en W .

Estas definiciones son válidas de forma general para cualquier tipo de robot, aunque son directamente aplicables para un objeto rígido que se mueve libremente en su espacio de trabajo. Para particularizarlas a robots articulados con un punto fijo es necesario realizar algunas consideraciones adicionales, lo que se efectuará en secciones posteriores.

2.1.1. Casos particulares

Con el objeto de facilitar la comprensión de las definiciones previas, se va a estudiar la representación del espacio de las configuraciones para tres robots

móviles, con distinta forma geométrica y diferente dimensión.

- Sea \mathbf{A} una bola de dimensión n que no tiene ninguna marca sobre su contorno. No es posible distinguir entre dos configuraciones del robot que sólo se diferencian en una rotación alrededor del centro de la bola. Por tanto, eligiendo O_A en el centro de la bola, el espacio de las configuraciones se puede representar por el espacio Euclídeo R^n .

- Sea \mathbf{A} un cilindro de longitud finita en R^3 . Este engloba el caso donde \mathbf{A} es un segmento de línea en R^3 . No es posible distinguir entre dos configuraciones que solamente se diferencian en una rotación alrededor del eje del cilindro. Entonces, eligiendo O_A sobre el eje de \mathbf{A} y uno de los ejes de F_A a lo largo del eje de \mathbf{A} , C se puede representar como $R^3 \times S^2$, donde S^2 denota la esfera unidad en R^3 . Por tanto, C es de dimensión 5.

- Sea \mathbf{A} un objeto con unas determinadas dimensiones que puede trasladarse pero no rotar. Entonces C se puede representar como R^n , donde una configuración q se define por las coordenadas de O_A en F_W .

2.2. Obstáculos en el C-espacio para un Objeto Rígido Móvil

En general, W contiene obstáculos fijos que son regiones de R^n . \mathbf{B} denota tanto al obstáculo "físico como al subconjunto de R^n que lo representa. En lo sucesivo se considerará que los obstáculos son rígidos y que están fijos en W . Por tanto, cada punto de \mathbf{B} tiene una posición fija con respecto a F_W .

Definición 3 *El obstáculo \mathbf{B} en W se proyecta en C sobre la región*

$$\mathbf{CB} = \{q \in C / \mathbf{A}(q) \cap \mathbf{B} \neq \emptyset\}$$

\mathbf{CB} se denomina obstáculo en el espacio de las configuraciones o ζ -obstáculo".

Esta es la definición propuesta por Lozano-Pérez [?] y es admitida de forma unánime en la bibliografía. En concreto, significa que si $q \in \mathbf{CB}$ entonces $\mathbf{A}(q)$ intersecta con \mathbf{B} . Por tanto, un ζ -obstáculo representa el conjunto de configuraciones de \mathbf{A} que producen colisión con \mathbf{B} . Los ζ -obstáculos son las restricciones en el movimiento del robot debidas a la presencia de obstáculos en el espacio de trabajo.

Definición 4 *El conjunto*

$$\mathbf{C}_{\text{free}} = \{q \in C / \mathbf{A}(q) \cap \mathbf{B} = \emptyset\}$$

se denomina espacio libre. Una configuración en \mathbf{C}_{free} se denomina una configuración libre de colisiones.

2.2.1. Ejemplos

A continuación se ilustrarán con ejemplos sencillos las nociones y definiciones establecidas previamente. En concreto, se mostrará una representación gráfica del C-obstáculo que se obtiene cuando se considera un robot móvil, con tres formas geométricas diferentes, cuyo espacio de trabajo se encuentra ocupado por un único obstáculo.

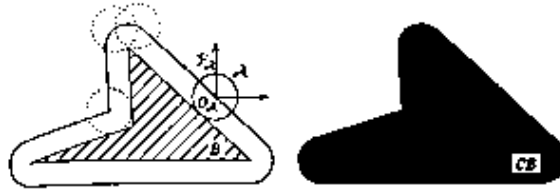


Figura 2.2: C-obstáculo para un disco móvil y un obstáculo poligonal

- Sea \mathbf{A} un robot con forma de disco que se mueve libremente en $W = \mathbb{R}^2$, donde existe un obstáculo \mathbf{B} poligonal como se muestra en la figura 2.2. En W se elige un sistema de referencia F_W . En \mathbf{A} se establece un sistema de referencia F_A que se mueve con el robot, cuyo origen O_A se sitúa en el centro del disco. En este caso una configuración q vendría dada por la posición del punto de referencia O_A respecto de F_W . Entonces el espacio de las configuraciones sería $C = \mathbb{R}^2$. El C-obstáculo \mathbf{CB} debido al polígono \mathbf{B} se obtiene aumentando \mathbf{B} de forma isotrópica con el radio de \mathbf{A} (figura 2.2). El límite de \mathbf{CB} es la curva seguida por el punto O_A cuando \mathbf{A} da vueltas sobre el contorno de \mathbf{B} . De esta forma se tienen definidas todas las posibles configuraciones del robot \mathbf{A} que producen colisión con el polígono \mathbf{B} .

- Sea \mathbf{A} un polígono convexo, en concreto un triángulo rectángulo (figura 2.3), que se mueve con una orientación fija (no puede rotar) en $W = \mathbb{R}^2$. En este es-

pacio de trabajo se encuentra un obstáculo **B** poligonal convexo, un rectángulo. El sistema de referencia F_A se elige rígidamente unido al robot, con su origen O_A situado en uno de los vértices de **A**. Debido a que el robot se mueve con una orientación fija, una configuración vendría dada por la posición de O_A . Por tanto, $C = R^2$. En la figura 2.3 se muestran los C-obstáculos en R^2 obtenidos para dos diferentes orientaciones fijas de **A**. El límite de cada C-obstáculo es la curva seguida por O_A cuando **A** se desplaza en contacto con el contorno de **B**, sin que los interiores de **A** y **B** se solapen. Cada C-obstáculo es un polígono convexo.

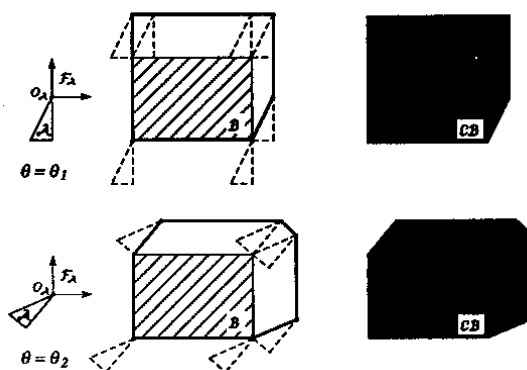


Figura 2.3: C-obstáculo para un triángulo móvil con orientación fija y un obstáculo poligonal

- Sea **A** un polígono convexo (figura ??) que se puede trasladar y rotar en $W = R^2$, donde existe un obstáculo **B** poligonal convexo. Con la misma elección de F_A que la efectuada en el caso previo, una configuración estaría dada por la posición (x, y) y la orientación θ de F_A con respecto F_W . Por tanto, C sería un espacio de dimension 3. La representación de C vendría dada parametrizando cada configuración q por $(x, y, \theta) \in R^2 \times [0, 2\pi]$. La figura ?? muestra el C-obstáculo **CB**, donde cada corte transversal de **CB**, perpendicular al eje θ , se corresponde con un **CB** para una orientación determinada de **A**, como apareció en el caso previo.

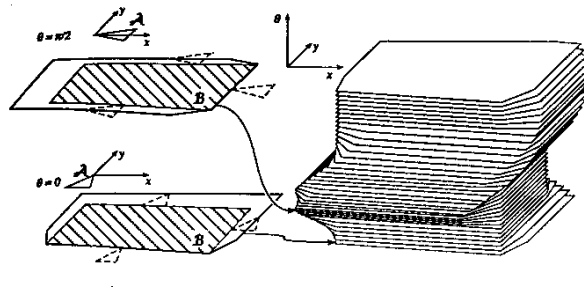


Figura 2.4: C-obstáculo para un triángulo móvil con orientación variable y un obstáculo poligonal

2.3. Espacio de las Configuraciones para un Robot Articulado

En las definiciones presentadas previamente se estaba considerando un objeto rígido móvil, en un espacio de trabajo bidimensional o tridimensional. Estas nociones se pueden extender para robots formados por varios objetos rígidos, que se mueven, conectados por articulaciones mecánicas que imponen unas restricciones a su movimiento relativo. Un ejemplo típico es un brazo manipulador, que consta de una secuencia de objetos rígidos conectados en una cadena mediante articulaciones.

Sea \mathbf{A} un robot articulado formado por p objetos rígidos o *elementos* $\mathbf{A}_1, \dots, \mathbf{A}_p$ que se mueven. Dos objetos \mathbf{A}_i y \mathbf{A}_j cualesquiera podrían estar conectados por una articulación. Se supondrá, para simplificar, que la articulación es de revolución o prismática. Existen otros tipos de articulaciones, aunque éstas dos son las más típicas. Una articulación de revolución restringe el movimiento relativo de \mathbf{A}_i y \mathbf{A}_j a una rotación alrededor de un eje fijo con respecto a ambos elementos. Una articulación prismática es una conexión que limita el movimiento relativo a una traslación a lo largo de un eje fijo con respecto a ambos objetos.

Sea F_{A_i} el sistema de referencia unido a \mathbf{A}_i , $i \in [1, p]$. Por la Definición 1 en la página 22, la configuración de \mathbf{A} es una especificación de la posición y la orientación de cada sistema de referencia F_{A_i} , $i \in [1, p]$, con respecto a F_W . Si los distintos objetos pudieran moverse de forma independiente en $W = R^n$, el espacio

de las configuraciones de \mathbf{A} sería $C' = \underbrace{(R^n \times SO(N)) \times \cdots \times (R^n \times SO(N))}_p$, donde $SO(N)$ es el Grupo Ortogonal Especial de matrices $N \times N$ con columnas y filas ortonormales y determinante $+1$. Sin embargo, las distintas articulaciones imponen restricciones en las configuraciones posibles de C' . Estas restricciones seleccionan un subespacio C de C' de dimensión más pequeña, que es el espacio de las configuraciones de \mathbf{A} .

Ejemplo.- En la figura ?? se muestra un manipulador planar con dos articulaciones. Está formado por dos objetos \mathbf{A}_1 y \mathbf{A}_2 . El primer elemento está conectado a la base por una articulación de revolución; \mathbf{A}_2 está unido a \mathbf{A}_1 mediante una articulación prismática. El espacio de trabajo es $W = R^2$. Considerando que los objetos fueran independientes, $C' = (R^2 \times SO(2))^2$ es un espacio de dimensión seis. La primera articulación impone dos restricciones: el punto sobre el que gira \mathbf{A}_1 está fijo en el espacio de trabajo. La segunda articulación impone dos restricciones más en la posición y orientación de \mathbf{A}_2 respecto a \mathbf{A}_1 . Las cuatro restricciones son independientes y determinan un subespacio C de dimensión dos en C' .

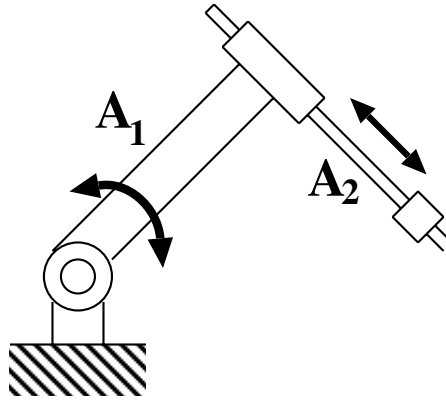


Figura 2.5: Manipulador planar con una articulación de revolución y otra prismática

En el resto de esta sección, se supondrá que \mathbf{A} es un robot articulado cuya cadena cinemática no está cerrada.

Con esta suposición, sean dos elementos \mathbf{A}_i y \mathbf{A}_j de un manipulador formado por p elementos, tal que \mathbf{A}_j se encuentra unido a \mathbf{A}_i , siendo $j > i$. Se puede

definir una configuración de \mathbf{A}_j respecto a \mathbf{A}_i especificando la posición y la orientación de F_{A_j} con respecto a F_{A_i} . Se supone, inicialmente, que no existen topes mecánicos en las articulaciones. Se denota con $C_j^{(i)}$ al C-espacio de \mathbf{A}_j respecto a \mathbf{A}_i . Si dos objetos están conectados por una articulación de revolución se tiene que $C_j^{(i)} = S^1$, donde S^1 denota al círculo unidad en R^2 . Si los dos elementos están unidos por una articulación prismática entonces se tiene que $C_j^{(i)} = R$. Por tanto, el espacio de las configuraciones de un robot articulado \mathbf{A} formado por p elementos rígidos conectados por p_1 articulaciones de revolución y p_2 prismáticas ($p = p_1 + p_2$) es:

$$C = \underbrace{S^1 \times \cdots \times S^1}_{p_1} \times \underbrace{R \times \cdots \times R}_{p_2}$$

El C-espacio del manipulador planar de la figura ?? es $S^1 \times R$.

Un problema práctico que se presenta es que la dimensión del espacio de las configuraciones aumenta con el número de articulaciones.

El C-espacio de las configuraciones de un robot articulado con p articulaciones prismáticas o de revolución es una variedad de dimensión p ([?], [?]). Una carta de esta variedad se puede definir simplemente asociando un ángulo, entre 0 y 2π , con cada articulación de revolución y un número real con cada articulación prismática. Entonces, una configuración q se representa por una lista (q_1, \dots, q_p) de coordenadas, donde cada una describe la posición y la orientación relativa de dos elementos unidos en \mathbf{A} . Los robots con articulaciones de revolución tienen espacios de las configuraciones conectados de forma múltiple y son necesarias varias cartas para formar un atlas. Sin embargo, en la práctica, basta con considerar una única carta para cada coordenada angular perteneciendo al intervalo $[0, 2\pi)$, y aplicar aritmética módulo 2π para cada coordenada.

Ejemplo.- Sea el manipulador planar con dos articulaciones de revolución de la figura ?? (a). Su espacio de las configuraciones sería $S^1 \times S^1$, por tanto un toro en el espacio Euclídeo de tres dimensiones. Una carta en esta variedad se puede definir asociando un ángulo en $(0, 2\pi)$ con cada una de las dos articulaciones. La figura ??(b) muestra estos dos ángulos θ_1 y θ_2 . El toro está conectado de forma múltiple y se pueden necesitar varias cartas para formar un atlas. Sin embargo, en muchos casos, es adecuado considerar una única carta y permitir que los ángulos θ_1 y θ_2 varíen en $[0, 2\pi)$, con aritmética módulo 2π . Esta simplificación

corresponde con representar el toro por un cuadrado $[0, 2\pi] \times [0, 2\pi]$ (figura ??(c)).

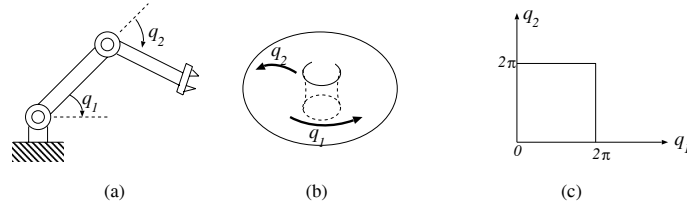


Figura 2.6: Manipulador planar de revolución y su C-espacio

2.4. Obstáculos en el C-espacio para un Robot Articulado

Existen dos tipos de C-obstáculos para un robot articulado:

1. Los correspondientes a la colisión de un elemento \mathbf{A}_i con los obstáculos en su espacio de trabajo.
2. Los originados por la colisión entre dos elementos, \mathbf{A}_i y \mathbf{A}_j , del robot.

En [?] se proponen las siguientes definiciones para estas dos clases de C-obstáculos:

Definición 5 *El C-obstáculo debido a la interacción de un elemento \mathbf{A}_i con un obstáculo \mathbf{B}_j se denota por \mathbf{CB}_{ij} y se define por*

$$\mathbf{CB}_{ij} = \{q = (q_1, \dots, q_i, \dots, q_p) \in C/\mathbf{A}_i(q_1, \dots, q_i) \cap \mathbf{B}_j \neq \emptyset\}$$

Definición 6 *El C-obstáculo debido a la interacción del elemento \mathbf{A}_i con el elemento \mathbf{A}_j , siendo $i < j$, se denota por \mathbf{CA}_{ij} y se define por*

$$\mathbf{CA}_{ij} = \{q = (q_1, \dots, q_i, \dots, q_j, \dots, q_p) \in C/\mathbf{A}_i(q_1, \dots, q_i) \cap \mathbf{A}_j(q_1, \dots, q_j) \neq \emptyset\}$$

2.4.1. Ejemplo

Sea \mathbf{A} un robot articulado formado por dos elementos, \mathbf{A}_1 y \mathbf{A}_2 , que son segmentos de rectas y que están conectados por una articulación de revolución.

El primer elemento está unido a la base mediante otra articulación de revolución. El robot se mueve en $W = R^2$, donde existe un conjunto \mathbf{B} de obstáculos puntuales. Se eligen como sistemas de referencia F_W en W , F_{A_1} unido a \mathbf{A}_1 y F_{A_2} unido a \mathbf{A}_2 , como se muestra en la figura ???. Una configuración q vendría dada por la orientación relativa de dos elementos unidos, siendo en este caso $q = (\theta_1, \theta_2)$. Entonces el espacio de las configuraciones sería $C = S^1 \times S^1$.

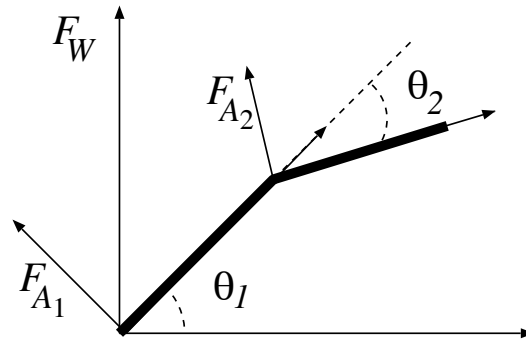


Figura 2.7: Manipulador planar de revolución

Los C -obstáculos que se obtienen debidos a los obstáculos puntuales (figura ??) situados en el espacio de trabajo aparecen en la figura ???. Un obstáculo puntual en W se transforma en una curva en C , que representa todas las posibles colisiones del obstáculo con la longitud completa del robot.

El cálculo de los \mathbf{CA}_i se puede realizar adaptando los métodos que se presentarán para robots móviles. De todas formas, únicamente sería necesario calcularlos una sola vez para un determinado robot articulado. Sin embargo, los C -obstáculos \mathbf{CB}_i es necesario calcularlos cada vez que su espacio de trabajo se modifique. Realmente esto último conlleva una gran carga computacional que es necesario optimizar, y es el objetivo fundamental de los diferentes trabajos de investigación dedicados a este tema, entre los que se encuentra el presentado en esta memoria.

Capítulo 3

Revisión de Métodos para el Cálculo de los C-obstáculos

En el capítulo previo se han definido los conceptos de espacio de las configuraciones y de C-obstáculo y se han propuesto ejemplos donde, intuitivamente, se ponía de manifiesto la forma de generar un C-obstáculo. Además, en la introducción se presentaba una revisión, a grandes rasgos, de los principales trabajos que han contribuido al avance en la representación de los C-obstáculos. Con ella se pretendía proporcionar una visión global de la evolución de las distintas técnicas implicadas.

En las secciones siguientes se van a analizar, de forma detallada, los métodos utilizados por diferentes autores para la generación de los C-obstáculos. En primer lugar, se revisarán los procedimientos para robots móviles y, en segundo lugar, para articulados con un punto fijo. Como se podrá comprobar la mayoría de los trabajos, principalmente en sus orígenes, se centran en robots móviles. A continuación, surgieron distintos procedimientos donde se extendían estas técnicas a articulados y, posteriormente, se propusieron nuevos métodos especialmente pensados para este tipo de robot. En general, existen pocos trabajos de investigación dedicados a articulados trabajando en espacios tridimensionales.

Los detractores del espacio de las configuraciones aducen que la principal dificultad que esta idea plantea es proyectar los obstáculos en ese espacio de una forma rápida, eficiente y exacta. El análisis que se presenta a continuación servirá

para mostrar cómo se ha intentado salvar estas dificultades. Con él se pretende, además, contrastar las diferentes técnicas surgidas con la propuesta en esta tesis doctoral y, de esta forma, poner de manifiesto las principales aportaciones y mejoras introducidas.

3.1. Métodos para Calcular los C-obstáculos de Robots Móviles

En esta sección se van a examinar las diferentes técnicas utilizadas para construir los C-obstáculos, para un objeto rígido que se mueve libremente sobre un plano ocupado por obstáculos. Solamente se considerará que los objetos en el espacio de trabajo, tanto obstáculos como el robot, están representados por modelos algebraicos. Estos modelos presentan como principal ventaja que los objetos se pueden describir por un número pequeño de parámetros y, además, incluyen un caso particular muy interesante, como es la representación mediante polígonos o poliedros.

En [?] se presenta una recopilación de los principales trabajos que abordan este problema con la construcción de dos representaciones diferentes de los C-obstáculos, denominadas, respectivamente, *representación de un C-obstáculo con C-restricciones* y *representación límite de un C-obstáculo*. En el análisis se muestra cómo en ambas la complejidad computacional depende de la geometría de los objetos, en concreto, del número de vértices. Además, los algoritmos propuestos para las dos representaciones inicialmente tienen como limitación que tanto el robot como los obstáculos sean convexos. Posteriormente, se considera que los objetos no convexos se pueden descomponer en una unión de convexos, lo que supone una carga computacional adicional.

La primera utiliza la lógica como herramienta, de forma que define un predicado CB que toma una configuración q como argumento. $CB(q)$ toma valor verdadero si y sólo si $q \in \mathbf{CB}$, donde \mathbf{CB} es el C-obstáculo representado por CB. Este es el resultado de un teorema propuesto y demostrado en [?] para espacios de trabajos poligonales, donde se propone una determinada expresión para el predicado CB. Posteriormente, en [?] se propone una ampliación de esta expresión que recoge la extensión a espacios de trabajos poliédricos. Otras

expresiones del predicado \mathbf{CB} han sido propuestas en [?].

La segunda representación es una descripción explícita del límite del C-obstáculo \mathbf{CB} , como una lista de caras, lados y vértices, con sus ecuaciones y su relación topológica de adyacencia. En [?] se proponen dos métodos para construir el C-obstáculo, uno con mayor tiempo de ejecución que el otro, donde tanto \mathbf{A} como \mathbf{B} pueden ser polígonos o poliedros. En [?] se describe otro método que directamente construye el límite de \mathbf{CB} a partir de la descripción de \mathbf{A} y \mathbf{B} , primero para polígonos y después para poliedros.

A continuación se van a analizar, de forma detallada, las dos representaciones. En primer lugar, se considerará que el espacio de trabajo es bidimensional y los objetos se modelan como regiones poligonales. Para esta situación, previamente, es necesario distinguir entre dos tipos de contactos entre el robot y el obstáculo, que conducen a dos condiciones de aplicabilidad; una para cada contacto. Ambas condiciones se utilizarán en las dos representaciones. Posteriormente, el análisis se extenderá para espacios de trabajo poliédricos.

3.1.1. Espacio de trabajo poligonal

En esta subsección se examina el caso donde \mathbf{A} es un polígono, que se puede mover libremente en el plano entre obstáculos modelados como regiones poligonales. Se supone que tanto el robot como los obstáculos son convexos. Para un obstáculo dado \mathbf{B} , se han de construir las ecuaciones que definen las superficies que limitan al correspondiente C-obstáculo \mathbf{CB} . Estas superficies provienen de dos tipos de contactos ([?] [?]) entre \mathbf{A} y \mathbf{B} :

1. contacto tipo A, cuando un lado de \mathbf{A} contiene un vértice de \mathbf{B}
2. contacto tipo B, cuando un vértice de \mathbf{A} está contenido en un lado de \mathbf{B}

Si la zona de contacto $\mathbf{A}(q) \cap \mathbf{B}$ contiene un vértice de \mathbf{A} y uno de \mathbf{B} , entonces el tipo de contacto es A y B.

La suposición de que los interiores de \mathbf{A} y \mathbf{B} no intersecten implica que el contacto tipo A solamente es posible para un conjunto de configuraciones de \mathbf{A} . Este conjunto está determinado por dos expresiones, con cuya conjunción se define la denominada condición de aplicabilidad del contacto tipo A entre el lado j de \mathbf{A} y el vértice i de \mathbf{B} , denotada por $\text{APPL}_{i,j}^{\mathbf{A}}(q)$. Esta condición

establece la posibilidad de que se produzca un contacto tipo A, para lo que el lado j de A se debe encontrar en una posición adecuada respecto al vértice i de B. Si A se desplaza de forma que se mantenga el contacto tipo A, entonces la configuración del robot se mueve a lo largo de una superficie del C-espacio, denominada C-superficie de tipo A, cuya ecuación es $f_{i,j}^{\mathbf{A}}(q) = 0$. Esta superficie divide C en dos semiespacios. El C-obstáculo CB se encuentra completamente dentro del semiespacio determinado por $f_{i,j}^{\mathbf{A}}(q) \leq 0$.

De la misma forma, un contacto tipo B sólo se produce para un subintervalo de orientaciones de A determinado por dos condiciones. La conjunción de estas dos condiciones define la condición de aplicabilidad del contacto tipo B entre el vértice i de A y el lado j de B, denotada por $APPL_{i,j}^{\mathbf{B}}(q)$. Cuando A se mueve manteniendo el contacto tipo B, la configuración del robot se mueve en una superficie, denominada C-superficie de tipo B, cuya ecuación es $f_{i,j}^{\mathbf{B}}(q) = 0$. El C-obstáculo CB se encuentra completamente dentro del semiespacio determinado por $f_{i,j}^{\mathbf{B}}(q) \leq 0$.

Las condiciones de aplicabilidad de los contactos tipo A y B sirven de base para las dos representaciones de los C-obstáculos que se analizan a continuación.

Representación del C-obstáculo con C-restricciones

Esta representación fue propuesta en [?] [?], y utiliza las condiciones de aplicabilidad para definir una C-restricción¹ de tipo A y otra de tipo B.

Definición 7 *La expresión*

$$APPL_{i,j}^{\mathbf{A}}(q) \Rightarrow [f_{i,j}^{\mathbf{A}}(q) \leq 0]$$

se denomina una C-restricción de tipo A y se denota por $CONST_{i,j}^{\mathbf{A}}(q)$.

Definición 8 *La expresión*

$$APPL_{i,j}^{\mathbf{B}}(q) \Rightarrow [f_{i,j}^{\mathbf{B}}(q) \leq 0]$$

se denomina una C-restricción de tipo B y se denota por $CONST_{i,j}^{\mathbf{B}}(q)$.

En esta representación se describe un C-obstáculo CB mediante un predicado $CB(q)$. $CB(q)$ se construye como la conjunción de todas las C-restricciones

¹C-constraint

de tipo A y B generadas por los lados y vértices de \mathbf{A} y \mathbf{B} . En [?] y [?] se demuestra el siguiente teorema:

Teorema 1 Sean \mathbf{A} y \mathbf{B} dos polígonos convexos. El C-obstáculo

$$\mathbf{CB} = \{q \in C / \mathbf{A}(q) \cap \mathbf{B} \neq \emptyset\}$$

es tal que

$$q \in \mathbf{CB} \iff \mathbf{CB}(q)$$

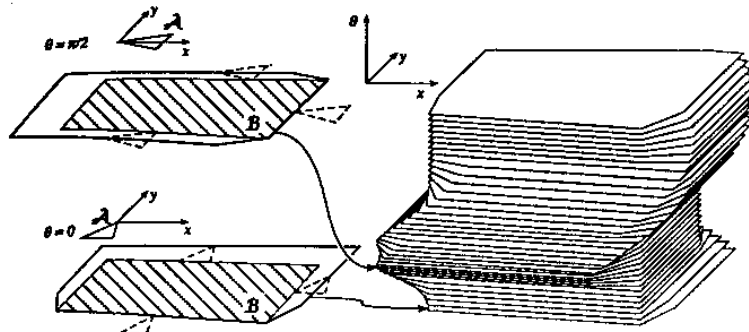
con

$$\mathbf{CB}(q) \equiv (\bigwedge_{i,j} \text{CONST}_{i,j}^{\mathbf{A}}(q)) \wedge (\bigwedge_{i,j} \text{CONST}_{i,j}^{\mathbf{B}}(q))$$

Utilizando el predicado \mathbf{CB} , se puede calcular si una configuración q se encuentra o no en \mathbf{CB} , evaluando individualmente cada C-restricción hasta que una de ellas evalúe a falso (entonces $q \notin \mathbf{CB}$) o todas evalúen a verdadero (entonces $q \in \mathbf{CB}$). El número de C-restricciones en $\mathbf{CB}(q)$ es $2n_A n_B$.

Si \mathbf{A} y \mathbf{B} son polígonos no convexos, se pueden representar como uniones finitas de polígonos convexos. Con esto, el predicado $\mathbf{CB}(q)$ se puede poner como la disyunción de los predicados asociados a cada una de las combinaciones posibles de los distintos polígonos convexos en los que se descomponen \mathbf{A} y \mathbf{B} . Esto supone la evaluación de un número mayor de predicados, además de introducir una carga computacional adicional asociada al algoritmo de descomposición de polígonos no convexos en una unión de convexos.

Cuando esta formulación general se particulariza para una determinada parametrización del C-espacio, que es diferente para cada tipo de robot, las expresiones para la evaluación del predicado $\mathbf{CB}(q)$ son de elevada complejidad aun en casos sencillos. En [?] se propone una expresión parametrizada de las C-restricciones para el caso de un robot que se mueve y gira libremente en un plano. Es interesante analizar la forma de los C-obstáculos. Para ello considérese un robot triangular que se mueve libremente en un espacio de trabajo ocupado por un obstáculo con forma rectangular (figura ??). Los C-obstáculos que se obtienen son volúmenes limitados por trozos de C-superficies. Estos trozos se denominan caras del C-obstáculo y cada una de ellas se corresponde con contactos tipo A o tipo B.

Figura 3.1: C-obstáculos cuando A y B son polígonos convexos

Representación límite del C-obstáculo

En la sección anterior se representaba un C-obstáculo mediante un predicado CB . Este concepto es conceptualmente simple, pero no proporciona una representación explícita del límite de un C-obstáculo CB , como una lista de caras, lados y vértices, con sus ecuaciones y su relación topológica de adyacencia. A continuación, se presenta un método para construir una representación del límite de CB cuando el robot solamente se traslada. Después, se describirá un método más general que se aplica cuando el robot se traslada y rota.

- Caso de traslación

Inicialmente se supone que A y B son convexos y que A se traslada con una orientación fija θ_0 . Se puede demostrar fácilmente que los vértices de CB_{θ_0} se obtienen de la forma siguiente:

1. Si se cumple la condición de aplicabilidad de un contacto tipo A $(APPL)_{i,j}^A(\theta_0)$ entonces los puntos

$$b_j - a_i(0, 0, \theta_0) \text{ y } b_j - a_{i+1}(0, 0, \theta_0)$$

son vértices de CB_{θ_0} , donde b_j y a_i son, respectivamente, los vértice B y A .

2. Si se cumple la condición de aplicabilidad de un contacto tipo B $(APPL)_{i,j}^B(\theta_0)$ entonces los puntos

$$b_j - a_i(0, 0, \theta_0) \text{ y } b_{j+1} - a_i(0, 0, \theta_0)$$

son vértices de \mathbf{CB}_{θ_0} .

En [?] se propone un algoritmo para construir \mathbf{CB}_{θ_0} , donde básicamente se establece cuándo se cumplen ambas condiciones de aplicabilidad. El algoritmo se basa en situar los vectores normales a los n_B lados de \mathbf{B} ($v_j^{\mathbf{B}}$) y los antinormales a los n_A lados de \mathbf{A} ($-v_i^{\mathbf{A}}(\theta_0)$) sobre un círculo de radio unidad. Así, establece que $(APPL)_{i,j}^{\mathbf{A}}(\theta_0)$ se cumple si y sólo si el vector $-v_i^{\mathbf{A}}(\theta_0)$ apunta entre $v_{j-1}^{\mathbf{B}}$ y $v_j^{\mathbf{B}}$, es decir, es una combinación lineal positiva de $v_{j-1}^{\mathbf{B}}$ y $v_j^{\mathbf{B}}$. De la misma forma $(APPL)_{i,j}^{\mathbf{B}}(\theta_0)$ se cumple si y sólo si $v_j^{\mathbf{B}}$ apunta entre $-v_{i-1}^{\mathbf{A}}(\theta_0)$ y $-v_i^{\mathbf{A}}(\theta_0)$. De esta forma se construyen los $n_A + n_B$ vértices de \mathbf{CB}_{θ_0} . El límite completo de \mathbf{CB}_{θ_0} se puede encontrar en un tiempo $O(n_A + n_B)$, que es suficientemente óptimo.

También, en [?] se propone otro algoritmo, un poco menos eficiente, para construir \mathbf{CB}_{θ_0} . En concreto, demuestra que

$$\mathbf{CB}_{\theta_0} = \text{conv}(\text{vert}(\mathbf{B}) \ominus \text{vert}(\mathbf{A}(0, 0, \theta_0)))$$

donde $\text{vert}(P)$ denota el conjunto de vértices de un polígono P , y $\text{conv}(S)$ denota la envolvente convexa (el polígono convexo más pequeño que encierra a S) de un conjunto de puntos S . Por tanto, en este artículo se propone el cálculo del C-obstáculo como la envolvente convexa del conjunto de puntos que resulta de efectuar la diferencia entre el conjunto de vértices de \mathbf{B} y el de $\mathbf{A}(0, 0, \theta_0)$. El cálculo de la envolvente convexa de un conjunto de puntos es un problema muy estudiado en Geometría Computacional. Los algoritmos propuestos tienen un tiempo de ejecución de $O(n \log n)$, siendo n el número de puntos del conjunto. Por tanto, el tiempo de ejecución de ese algoritmo es $O(n_A n_B \log(n_A n_B))$.

Si \mathbf{A} y \mathbf{B} son polígonos no convexos, se descomponen en polígonos convexos y se calcula el C-obstáculo correspondiente a cada par de polígonos. De la unión de estos C-obstáculos se obtiene \mathbf{CB}_{θ_0} que es un polígono. En [?] se propone un método para la representación del límite de \mathbf{CB}_{θ_0} , cuando se consideran polígonos no convexos, en un tiempo $O(n_A^2 n_B^2 \log n_A n_B)$. Utilizando un algoritmo de línea de barrido, divide \mathbf{A} y \mathbf{B} en triángulos y trapecios. Para cada par de polígonos se calcula el límite del \mathbf{CB} correspondiente utilizando el método presentado anteriormente para polígonos convexos y, después, se realiza la unión de todos ellos.

- Caso General

En este caso se supone que \mathbf{A} puede trasladarse y girar. Si los objetos son polígonos convexos, el método propuesto para calcular el límite de \mathbf{CB}_{θ_0} en un tiempo $O(n_A + n_B)$ se puede extender para calcular las caras, los lados y los vértices del límite de \mathbf{CB} , y su relación de adyacencia. Cada cara corresponde a un contacto tipo A o tipo B. La intersección de la cara con un plano de θ constante es un segmento de línea. Las coordenadas de sus puntos finales son funciones de θ . Estas funciones definen las ecuaciones de dos de los lados que limitan la cara. La sección transversal a través de \mathbf{CB} con la misma orientación también permite calcular la relación de adyacencia de la cara con otras caras para esa orientación. Desplazando un plano a θ constante desde $\theta = 0$ hasta $\theta = 2\pi$ y deteniéndose en cada orientación crítica (donde un lado de \mathbf{A} está alineado con uno de \mathbf{B}), se puede construir una representación explícita del límite de \mathbf{CB} en un tiempo $O(n_A n_B (n_A + n_B))$. Desplazando un plano a θ constante desde $\theta = 0$ hasta $\theta = 2\pi$, se puede construir una representación explícita del límite de \mathbf{CB} en un tiempo $O(n_A n_B (n_A + n_B))$.

Si \mathbf{A} o \mathbf{B} son no convexos entonces se puede pensar en descomponerlos en partes convexas, calcular los límites de los C-obstáculos correspondientes y obtener su intersección. Sin embargo, la intersección de los límites que están formados por caras de tipo A o B no es fácil. En lugar de hacer esto, se podría calcular el límite de la sección transversal del C-obstáculo para una orientación fija de \mathbf{A} , y analizar cómo varía este límite con los cambios de orientación de \mathbf{A} . Pero, a diferencia con el caso convexo, la sección transversal puede experimentar cambios topológicos significativos en orientaciones distintas a aquellas donde un lado de \mathbf{A} es paralelo a un lado de \mathbf{B} . En [?] se presenta un método donde directamente se construye el límite de \mathbf{CB} (es decir, el límite del C_{free}) a partir de \mathbf{A} y \mathbf{B} . En este cálculo se emplea un tiempo $O(n_A^3 n_B^3 \log n_A n_B)$. Esta representación describe el límite del C_{free} como una colección de trozos de C-superficies, denominadas caras, y sus relaciones de adyacencia.

Un método más general que no diferencia entre polígonos convexos y no convexos fue propuesto por [?]. Este algoritmo acepta como entrada dos polígonos, donde uno permanece fijo y el otro puede desplazarse y girar, y calcula una representación del correspondiente C-obstáculo, incluyendo información sobre

los contactos. No se construye el C-obstáculo tridimensional de forma completa, sino que se calculan múltiples proyecciones sobre planos. Con ello el algoritmo genera una estructura de datos que contiene toda la información métrica y topológica necesaria para describir el conjunto de configuraciones coherentes con cada condición de contacto que puede presentarse. El procedimiento, primero, genera regiones en el C-espacio correspondientes a contactos entre un vértice y un lado y establece relaciones locales de vecindario. A continuación, tiene en cuenta las características geométricas no locales que restringen el conjunto de configuraciones posibles para un determinado contacto. Con ello, fija los límites de conflicto para cada región y, posteriormente, elimina los límites extraños obteniendo regiones que están de acuerdo con la geometría de contacto local y no local. El procedimiento finaliza eliminando regiones globalmente imposibles. La complejidad del algoritmo es $O(n_A^3 n_B^3)$.

3.1.2. Espacio de trabajo poliédrico

En esta sección se extienden algunos de los resultados presentados previamente para el caso de espacios de trabajo poliédricos. El C-obstáculo \mathbf{CB} correspondiente a un obstáculo \mathbf{B} es, en este caso, una región de seis dimensiones de C limitada por C-superficies de dimensión cinco. Para este caso, las C-superficies provienen de tres tipos de contactos entre \mathbf{A} y \mathbf{B} :

1. contactos tipo A entre una cara de \mathbf{A} y un vértice de \mathbf{B}
2. contactos tipo B entre un vértice de \mathbf{A} y una cara de \mathbf{B}
3. contactos tipo C entre un lado de \mathbf{A} y un lado de \mathbf{B}

Si se utiliza la representación de un C-obstáculo mediante C-restricciones, hay que tener en cuenta que cada tipo de contacto produce una C-restricción diferente. Así, existirán C-restricciones de tipo A, B y C generadas por las caras, vértices y lados de \mathbf{A} y \mathbf{B} . Como en el caso poligonal, si \mathbf{A} y \mathbf{B} son convexos, el predicado \mathbf{CB} definido por $q \in \mathbf{CB} \Leftrightarrow \mathbf{CB}(q)$ se puede construir como la conjunción de todas las C-restricciones [?]. Sin embargo, hay muchas más C-restricciones en el caso poliédrico que en el poligonal, lo que aumenta considerablemente el tiempo de cálculo. En [?] se propone otra expresión para el predicado \mathbf{CB} .

Si \mathbf{A} y \mathbf{B} son poliedros no convexos, al igual que para espacios de trabajo poligonales, se pueden descomponer en convexos. Como en el caso poligonal, $\text{CB}(q)$ llega a ser una disyunción de los predicados asociados a los poliedros convexos.

En cuanto al formalismo de la representación límite, existen varios algoritmos para calcular la representación explícita del límite de un C-obstáculo \mathbf{CB} cuando el robot \mathbf{A} solamente puede trasladarse. En [?], si \mathbf{A} y \mathbf{B} son poliedros convexos, se propone la siguiente expresión para \mathbf{CB}

$$\mathbf{CB}_{\Theta_0} = \text{conv}(\text{vert}(\mathbf{B}) \ominus \text{vert}(\mathbf{A}(0, \Theta_0)))$$

Existen algoritmos para calcular la envolvente convexa de un conjunto de puntos en un espacio de dimensión tres con un tiempo de ejecución de $O(n_A n_B \log(n_A n_B))$. Un algoritmo más eficiente ($O(n_A + n_B + n_A n_B)$) para construir el límite de \mathbf{CB}_{Θ_0} se encuentra en [?]. [?] proponen una adaptación de su método utilizado en espacios poligonales para calcular el límite de \mathbf{CB}_{Θ_0} , cuando \mathbf{A} y \mathbf{B} son poliedros limitados por caras triangulares o trapezoidales. El tiempo de cálculo es $O(n_A^3 n_B^3 \log(n_A n_B))$. El método se puede extender con la misma complejidad de tiempo asintótica para cualquier par de poliedros, descomponiendo sus caras en triángulos y trapecios.

3.1.3. Consideraciones finales

Para finalizar esta revisión detallada de las técnicas más representativas para la generación de los C-obstáculos para robots móviles, es necesario realizar dos consideraciones adicionales, que a la vez sirven como conclusiones de este apartado.

1. En todos los algoritmos presentados en esta sección, una característica importante a destacar es que su tiempo de computación depende, tanto si se trata de polígonos convexos o no convexos como de poliedros, del número de vértices del robot y de los obstáculos.
2. Una condición necesaria que se encuentra presente en todos ellos es que los vértices y lados de los obstáculos y del robot han de estar perfectamente identificados. Esto supone que existe, bien un conocimiento del espacio de

trabajo modelado mediante ecuaciones algebraicas, o bien un sistema de visión con suficiente precisión como para tener una representación exacta del entorno.

3.2. Métodos para Calcular los C-obstáculos de Robots Articulados

En esta sección se examinarán los diferentes trabajos de investigación dedicados a la construcción de los C-obstáculos para robots articulados. En estos trabajos se realizan tres suposiciones para simplificar la representación, que ya fueron expuestas en el capítulo previo, pero que es conveniente tener presente.

1. El robot está formado por una secuencia de p elementos \mathbf{A}_1 hasta \mathbf{A}_p , donde dos objetos sucesivos \mathbf{A}_i y $\mathbf{A}_{(i+1)}$ están conectados por una articulación prismática o una de revolución. \mathbf{A}_1 es el primer objeto de la secuencia y está conectado al espacio de trabajo por una articulación. Una configuración q de \mathbf{A} se representa por una lista de p parámetros, (q_1, \dots, q_p) . Cada parámetro q_i ($i \in [1, p]$) determina la posición y la orientación de F_{A_i} respecto a $F_{A_{i-1}}$, con $F_{A_0} = F_W$.
2. Cada articulación q_i varía en un intervalo limitado $I_i = (q_i^-, q_i^+)$. Si q_i corresponde con una articulación de revolución sin topes mecánicos, el intervalo es $I_i = [0, 2\pi)$ y se emplea aritmética módulo 2π .
3. Dos elementos \mathbf{A}_i y \mathbf{A}_j cualesquiera no pueden chocar entre sí. Por tanto, los únicos choques que se consideran son aquéllos entre los elementos \mathbf{A}_i ($i \in [1, p]$) y los obstáculos \mathbf{B}_j ($j \in [1, q]$).

También hay que tener en cuenta que la posición y la orientación de F_{A_i} respecto a F_W , para cualquier $i \in [1, p]$, está completamente determinada por los parámetros q_1 hasta q_i . Como consecuencia, se denota por $\mathbf{A}_i(q_1, \dots, q_i)$ a la región de W ocupada por el objeto \mathbf{A}_i .

En general los trabajos desarrollados para la construcción de la región del C-obstáculo para robots articulados se pueden agrupar en:

1. Los que aplican la idea del planteamiento de *descomposición aproximada en celdas* propuesta para robots móviles a robots articulados. Este consiste en decidir si las celdas generadas por la descomposición del C-espacio pertenecen o no a la región del C-obstáculo, utilizando las ecuaciones explícitas que definen los límites de los C-obstáculos. Para móviles estas ecuaciones se pueden establecer utilizando los métodos presentados en la sección previa. Este planteamiento podría ser directamente aplicable a articulados, pero las ecuaciones explícitas de los límites de los C-obstáculos son más difíciles de establecer y de explotar. En su lugar se utiliza un método que consiste en discretizar el intervalo de movimiento de cada articulación en intervalos más pequeños; después, se comprueba si el robot colisiona o no con los obstáculos cuando las distintas articulaciones se encuentran en estos subintervalos. Este método, con pequeñas variaciones, es el que se propone en [?], [?] y [?].

2. Los que calculan las ecuaciones exactas del límite del C-obstáculo. En estos trabajos se consideran las ecuaciones algebraicas que definen los obstáculos y las expresiones matemáticas que describen la cinemática del robot. A partir de las condiciones que establecen cuándo los elementos del robot intersectan con los obstáculos se calculan las ecuaciones explícitas del límite de los obstáculos en el espacio de las configuraciones. Este método, para manipuladores articulados concretos, se presenta en [?] y [?] y para cualquier tipo de manipulador general en [?] y [?].

En los siguientes apartados se van a analizar los diferentes algoritmos que han sido propuestos en cada uno de estos dos grupos, comenzando por los que se plantearon en primer lugar.

Existen distintos métodos para construir los C-obstáculos y la mayoría de ellos se centran en manipuladores cuyo espacio de trabajo es R^2 . Pocos contemplan manipuladores trabajando en espacios tridimensionales, como el PUMA, el SCARA, etc.

3.2.1. Descomposición aproximada en celdas

El principio general de la descomposición aproximada en celdas se puede aplicar a robots articulados para la construcción del espacio libre y la región del C-obstáculo. El planteamiento, sin embargo, es diferente debido a que para este tipo de robots las ecuaciones que definen los límites de los C-obstáculos son difíciles de obtener y de tratar. Por ello, en estos trabajos se sigue un método ligeramente diferente que consiste en discretizar el movimiento de cada articulación en intervalos pequeños. Esto permite calcular, de forma aproximada, la región de los C-obstáculos y construir un grafo de conectividad, que se utiliza para buscar un canal que conecte la configuración inicial a la final en etapa de planificación de movimientos.

En primer lugar se describe el procedimiento básico para construir una aproximación de la región del C-obstáculo en $I_1 \times \dots \times I_p \subset R^p$. Cada intervalo I_i , $i = 1$ hasta p , se divide en intervalos más pequeños que no se solapan δ_{i,k_i} , con $k_i = 1, 2, \dots$, hasta alguna resolución predeterminada. Si la región recorrida por \mathbf{A} en el espacio de trabajo, cuando (q_1, \dots, q_p) varía sobre la celda de tipo rectangular $\delta_{1,k_1} \times \dots \times \delta_{p,k_p}$, no interseca con alguno de los obstáculos, entonces $\delta_{1,k_1} \times \dots \times \delta_{p,k_p}$ forma parte del espacio libre C_{free} ; si no, se considera que pertenece a la región del C-obstáculo. Considerando todas las celdas, se construye una aproximación del espacio libre como una colección de pequeñas celdas de tipo rectangular. En la figura ?? aparece este sencillo procedimiento, donde se muestra la división de cada intervalos en subintervalos que producen colisión (rayado) o no. Para la etapa de planificación, se puede generar un grafo de conectividad que represente la relación de adyacencia entre estas celdas y, después, buscar un canal. Sin embargo, el método de planificación en conjunto no es muy eficiente en cuanto al tiempo de computación y espacio de memoria.

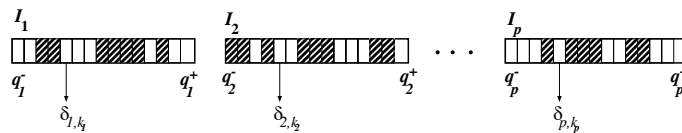


Figura 3.2: Construcción del C-espacio para un manipulador con p grados de libertad

A continuación se describe un método donde se introducen unas simples modificaciones que permiten reducir de manera considerable el número de celdas que tienen que ser examinadas de forma explícita. Se considera, inicialmente, el primer elemento \mathbf{A}_1 de \mathbf{A} . Su posición y orientación en W dependen sólo del valor de la coordenada q_1 . Si, para un valor dado de esta coordenada, interseca con un obstáculo, entonces no es necesario examinar los choques con los obstáculos del resto de los elementos, desde \mathbf{A}_2 hasta \mathbf{A}_p . Además, el trozo completo de dimensión $(p-1)$ de C que se proyecta en el valor dado de q_1 forma parte de la región del C-obstáculo. Posteriormente, se considera el elemento \mathbf{A}_i . De forma extendida, si para valores determinados de (q_1, \dots, q_i) , el elemento \mathbf{A}_i interseca con un obstáculo, el trozo completo de dimensión $(p-i)$ de C que se proyecta en los valores dados de q_1 hasta q_i forma parte de la región del C-obstáculo. Esto sugiere construir una representación del espacio de las configuraciones en forma de un árbol de p -niveles ([?] [?]), como se muestra en la figura ???. Los nodos a profundidad $i-1$ ($i \in [1, p]$) representan las descomposiciones del recorrido I_i en intervalos VACIOS y LLENOS. La raíz del árbol, es decir, el nodo a profundidad 0, representa el recorrido I_1 . En este nivel, los intervalos VACIOS de la descomposición están formados por todos los valores de q_1 para los que $\mathbf{A}_1(q_1)$ no choca con ningún obstáculo. Los intervalos LLENOS son el complementario de los VACIOS en I_1 . A continuación, los VACIOS se subdividen en intervalos más pequeños δ_{1,k_1} , con $k_1 = 1, 2, \dots$ hasta una determinada resolución. El árbol contiene un nodo a profundidad 1 para cada uno de esos intervalos. Se toma uno cualquiera de ellos δ_{1,k_1} . Su sucesor correspondiente representa una descomposición del recorrido I_2 en LLENOS y VACIOS. Los VACIOS están formados por todos los valores de q_2 para los que $\mathbf{A}_2(q_1, q_2)$ no interseca con los obstáculos cuando q_1 varía sobre δ_{1,k_1} . De nuevo, los VACIOS se descomponen en intervalos más pequeños δ_{2,k_2} y así sucesivamente. Excepto a profundidad $p-1$, los VACIOS se descomponen en intervalos más pequeños δ_{i,k_i} .

En la figura ??? se muestra el procedimiento propuesto en [?] para la construcción del C-espacio de un manipulador con tres articulaciones. Considerar el nodo a profundidad 2 marcado con '*' de la figura ???. Los intervalos VACIOS de este nodo están formados por todos los valores de q_3 tales que $\mathbf{A}(q_1, q_2, q_3)$ no choca con los obstáculos cuando q_1 varía sobre δ_{1,k_1} y q_2 varía sobre δ_{2,k_2} .

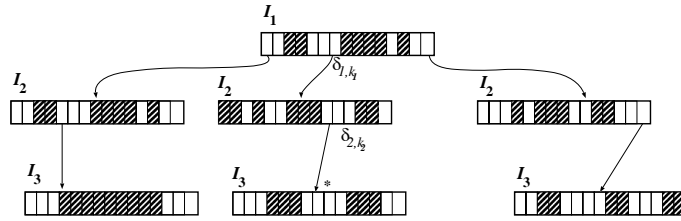


Figura 3.3: Procedimiento para la construcción del C-espacio de un manipulador

Debido a que se pueden clasificar trozos enteros como parte de la región del C-obstáculo en las primeras etapas del procedimiento, el tiempo y los requerimientos de memoria para construir el árbol son normalmente más pequeños que con la versión más simple del método presentada anteriormente.

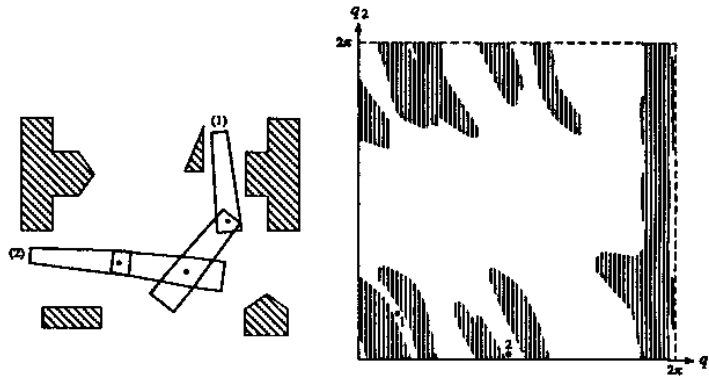


Figura 3.4: Manipulador planar de revolución: (a) Espacio de trabajo (b) C-obstáculos

En la figura ?? (a) se muestra un manipulador planar con dos articulaciones de revolución entre cinco obstáculos poligonales, en dos configuraciones libres marcadas respectivamente por (1) y (2). La figura ??(b) muestra la representación del espacio de las configuraciones del robot que se ha construido con el algoritmo descrito previamente. La región de los C-obstáculos se aproxima como un conjunto de trozos (slices) paralelos al eje q_2 . Los puntos designados por 1 y 2 representan las configuraciones del brazo marcadas con (1) y (2) en la figura (a).

El cálculo clave para construir el árbol de la fig ?? es determinar el recorrido

de los valores de $q_i (i \in [1, p])$ en el que $\mathbf{A}_i(q_1, \dots, q_{i-1}, q_i)$ no interseca con los obstáculos cuando q_1, \dots, q_{i-1} varían sobre los intervalos $\delta_{1,k_1}, \dots, \delta_{i-1,k_{i-1}}$.

En [?] se propone una técnica que consiste en fijar los valores de q_1, \dots, q_{i-1} , por ejemplo, en los puntos medios $q_{1,k_1}, \dots, q_{i-1,k_{i-1}}$ de los intervalos $\delta_{1,k_1}, \dots, \delta_{i-1,k_{i-1}}$. Entonces el cálculo se reduce a determinar los intervalos de valores de q_i para los que $\mathbf{A}_i(q_{1,k_1}, \dots, q_{i-1,k_{i-1}}, q_i)$ no interseca con los obstáculos cuando gira alrededor de un eje fijo (para una articulación de revolución) o cuando se traslada a lo largo de un eje fijo (para una articulación prismática).

Sin embargo, si se fijan los valores de q_1, \dots, q_{i-1} en lugar de permitir que varíen sobre los intervalos, tiene el inconveniente de generar una aproximación del espacio libre que no garantiza que sea conservadora. Una forma de abordar este inconveniente es calcular el desplazamiento euclídeo más largo de cualquier punto en \mathbf{A}_i para cualquier $q_i \in I_i$, cuando q_1, \dots, q_{i-1} varían sobre $\delta_{1,k_1}, \dots, \delta_{i-1,k_{i-1}}$. El objeto \mathbf{A}_i se aumenta de forma isotrópica con este desplazamiento y este objeto aumentado es el que se utiliza para descomponer el recorrido de I_i en intervalos VACIOS y LLENOS.

En [?] y [?] se presenta un procedimiento para mejorar la construcción del árbol que representa al C-espacio. El recorrido de I_i se descompone en intervalos clasificados como LLENOS, SEGURAMENTE VACIOS y POSIBLEMENTE VACIOS. Los LLENOS se definen de la misma forma que en el caso precedente. Los SEGURAMENTE VACIOS están formados por todos los valores de q_i tales que el objeto $\mathbf{A}_i(q_1, \dots, q_{i-1}, q_i)$ y la superficie barrida por los objetos $\mathbf{A}_{i+1}, \dots, \mathbf{A}_p$, cuando q_{i+1}, \dots, q_p varían sobre I_{i+1}, \dots, I_p , no intersecan con los obstáculos. Los POSIBLEMENTE VACIOS son los complementarios de los intervalos LLENOS y SEGURAMENTE VACIOS en I_i . Solamente los intervalos POSIBLEMENTE VACIOS se dividen en subintervalos más pequeños.

En [?] se propone optimizar el procedimiento asociando con cada intervalo POSIBLEMENTE VACIO en un nodo del árbol, una lista de obstáculos con los que chocan los elementos $\mathbf{A}_{i+1}, \dots, \mathbf{A}_p$. Posteriormente, cuando se consideran los sucesores de este nodo, solamente se tienen en cuenta los obstáculos en esta lista para examinar las posibles colisiones de \mathbf{A}_{i+1} . Entonces, en cada intervalo POSIBLEMENTE VACIO de I_{i+1} , se reduce la lista eliminando los obstáculos que no colisionan con la superficie barrida por $\mathbf{A}_{i+2}, \dots, \mathbf{A}_p$.

Para finalizar este análisis, hay que resaltar que todos los métodos basados en la descomposición presentan una gran carga computacional. Esto ha conducido a abordar otra filosofía donde se calculan las expresiones analíticas de los C-obstáculos para manipuladores.

3.2.2. Ecuaciones del límite de un C-obstáculo

Así como el método de la descomposición aproximada en celdas se aplica indistintamente a un brazo planar o espacial, los procedimientos que calculan las ecuaciones algebraicas de los C-obstáculos dependen en gran medida de los grados de libertad del brazo. En los dos apartados siguientes se analizan por separado los procedimientos más representativos que siguen esta vertiente, para robots planares, en primer lugar, y para espaciales, posteriormente.

Manipuladores articulados en el plano

Uno de los primeros análisis de la geometría de los límites de los C-obstáculos para distintos brazos planares fue presentado por Lumelsky en [?]. En ese trabajo el autor plantea que la transformación de los obstáculos del espacio de trabajo al espacio de las configuraciones está fuertemente relacionado con la cinemática inversa del mecanismo robótico. Además, para un manipulador planar con dos articulaciones de revolución y cuyos elementos son segmentos de líneas rectas (figura ??), propone utilizar un toro para representar su espacio de las configuraciones. Para este manipulador, la posición del efector final en coordenadas cartesianas viene dada por

$$x = l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2) \quad (3.1)$$

$$y = l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2) \quad (3.2)$$

Las dos configuraciones que puede que adoptar el manipulador para que el efector final esté en contacto con un obstáculo puntual situado en las coordenadas cartesianas (x, y) , se obtienen a partir de la resolución de las ecuaciones de la cinemática inversa

$$\theta_1 = \arctan\left(\frac{y}{x}\right) \pm \arccos\left(\frac{x^2 + y^2 + l_1^2 - l_2^2}{2l_1\sqrt{x^2 + y^2}}\right) \quad (3.3)$$

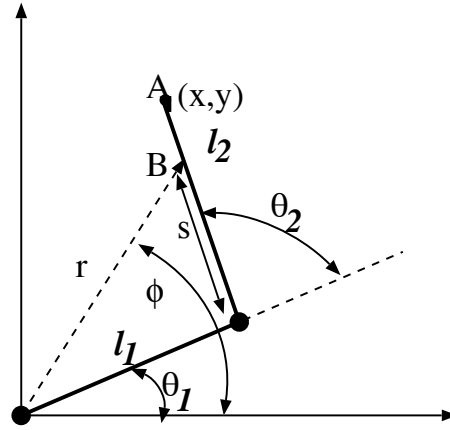


Figura 3.5: Manipulador planar con dos articulaciones de revolución

$$\theta_2 = \mp \arccos\left(\frac{x^2 + y^2 - l_1^2 - l_2^2}{2l_1l_2}\right) \quad (3.4)$$

Estas dos configuraciones, codo arriba y codo abajo, se corresponden respectivamente con la elección de los signos superiores e inferiores de las soluciones (\pm y \mp).

En [?] se propone simplificar estas expresiones utilizando las coordenadas polares del punto en lugar de coordenadas cartesianas, obteniéndose:

$$\theta_1 = \phi \pm \arccos\left(\frac{r^2 + l_1^2 - l_2^2}{2l_1r}\right) \quad (3.5)$$

$$\theta_2 = \mp \arccos\left(\frac{r^2 - l_1^2 - l_2^2}{2l_1l_2}\right) \quad (3.6)$$

donde

$$r = \sqrt{x^2 + y^2} \quad (3.7)$$

$$\phi = \arctan\left(\frac{y}{x}\right) \quad (3.8)$$

Para obtener la transformación de los obstáculos desde el espacio de trabajo al C-espacio hay que calcular todos los puntos de contacto a lo largo de toda la longitud del robot y no sólo con el efector final. Considerando las colisiones con el segundo elemento (con el primero son triviales), la solución no es única, sino que hay un conjunto infinito de soluciones parametrizadas por la variable s (ver figura ??). Así un obstáculo puntual en el espacio de trabajo se transforma en

una curva en el C-espacio, cuyas ecuaciones son:

$$\theta_1 = \phi \pm \arccos\left(\frac{r^2 + l_1^2 - s^2}{2l_1r}\right) \quad (3.9)$$

$$\theta_2 = \mp \arccos\left(\frac{r^2 - l_1^2 - s^2}{2l_1s}\right) \quad (3.10)$$

El signo doble que aparece en la expresión de θ_1 y θ_2 pone de nuevo de manifiesto las dos configuraciones, codo arriba y codo abajo, con las que el manipulador puede contactar con un punto. Además, las dos expresiones previas únicamente definen los choques del segundo elemento con el obstáculo puntual. Para puntos situados a una distancia $s < l_1$, hay que tener en cuenta las configuraciones con las que el primer elemento contacta con el obstáculo [?]. Concretamente

$$\theta_1 = \phi \quad (3.11)$$

$$\theta_2 = [\theta_{2_{min}}, \theta_{2_{max}}] \quad (3.12)$$

Un método general para obtener las ecuaciones del límite del C-obstáculo para manipuladores planares con n elementos se propone en [?]. Utiliza modelos poligonales para los elementos del robot y los obstáculos y determinan los límites de los obstáculos en el espacio imagen del robot. Después, los límites de los obstáculos son transformados de nuevo al espacio de las configuraciones del robot.

Más concretamente, el método consiste en utilizar el espacio imagen, donde se usa un cuaternión para representar el desplazamiento planar de un sistema de referencia desplazado respecto de uno fijo. El contacto entre un objeto poligonal con otro se representa por una subvariedad del espacio imagen denominada variedad de contacto. El movimiento de un elemento como parte de un robot se representa por otra subvariedad del espacio imagen denominada variedad alcanzable. La intersección de ambas variedades define la posición del efector final que simultáneamente satisface las condiciones de alcanzabilidad y de contacto. La parametrización de la intersección utilizando las variables de configuración de un robot proporciona la ecuación explícita para el límite del C-obstáculo. Este método se aplica en [?] a un robot planar 2R y en [?] a dos robots planares 2R cooperantes, para determinar las ecuaciones de los límites de los C-obstáculos.

Es interesante considerar la geometría de los obstáculos en el espacio de las

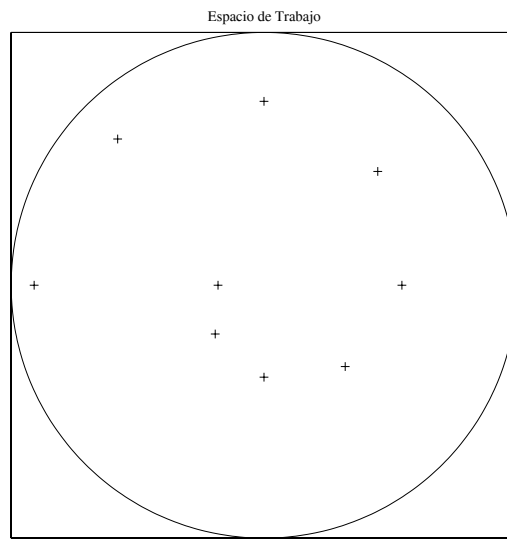


Figura 3.6: Obstáculos en el espacio de trabajo

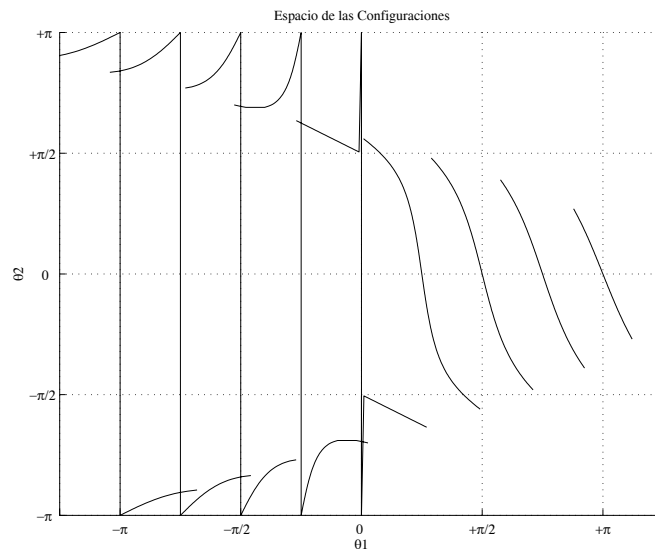


Figura 3.7: Obstáculos en el espacio de las configuraciones

configuraciones en función de su posición en el espacio de trabajo ([?] [?]). En la figura ?? se muestran nueve obstáculos puntuales que se encuentran en el espacio de trabajo de un manipulador planar de revolución. Están igualmente espaciados, con ϕ variando desde $-\pi$ hasta π con incrementos de $\pi/4$. Los cuatro

que se encuentran situados a mayor distancia radial del origen sólo pueden chocar con el segundo elemento, mientras que el resto pueden colisionar con ambos. En la figura ?? se muestran los nueve obstáculos en el espacio de las configuraciones.

- Los cuatro obstáculos cuya distancia radial es mayor que l_1 , sólo pueden colisionar con el segundo elemento y aparecen con forma de 'S' invertida.
- El resto (que pueden colisionar con ambos elementos) se muestran como líneas rectas, paralelas al eje θ_2 , con trazo discontinuo para los choques con el primer elemento y con trazo continuo para las colisiones con el segundo elemento.

En [?] y [?] se pone de manifiesto que el único parámetro que afecta a la curva del C-obstáculo es el radio r donde se encuentra el obstáculo puntual, mientras que el ángulo ϕ solamente introduce un desplazamiento en θ_1 . Entonces, se cumple que el C-obstáculo para un punto con coordenadas (r, ϕ) se puede obtener a partir del C-obstáculo de otro punto con coordenadas $(r, 0)$ desplazando en la dirección θ_1 una cantidad ϕ . Con esta propiedad, en [?] y [?] se propone construir mapas de primitivas del C-espacio para un manipulador dado almacenando el C-espacio generado a partir de obstáculos puntuales con $\phi = 0$ y diferentes valores de r . Este mapa de primitivas forman una familia de curvas en el C-espacio dependientes del parámetro r . Esta propiedad se cumple para manipuladores planares de revolución cuyos elementos están modelados con segmentos de línea y para otros con elementos poligonales. En [?] se obtienen las ecuaciones de la cinemática inversa para un manipulador con dos elementos poligonales. También en [?] y [?] se considera la extensión a obstáculos poligonales. En este caso la proyección de los obstáculos en el C-espacio viene representada por regiones limitadas por las mismas formas que para el caso de obstáculos puntuales y un manipulador con elementos modelados por rectas. Las curvas que limitan la región del C-obstáculo provienen de los contactos entre un punto (un vértice) y un segmento de línea.

Manipuladores articulados en el espacio

En [?], [?] y [?] se propone la extensión de sus algoritmos a robots espaciales, mediante una técnica denominada *slicing*. Esta es una variante del planteamiento de la descomposición aproximada en celdas. Consiste en descomponer el recorrido de uno de los grados de libertad en un número finito de intervalos. En cada intervalo, se calculan los contactos con los obstáculos considerando únicamente los grados de libertad restantes. Este planteamiento es aplicable directamente a estructuras mecánicas donde dos ejes de articulación intersecten.

Concretamente, en [?], [?] y [?] aplican el método a estructuras mecánicas tipo PUMA, aunque sólo analizan las tres primeras articulaciones, obteniendo C-espacios tridimensionales. Así, consideran que los dos últimos elementos trabajan en un plano, al igual que el manipulador planar con dos articulaciones de revolución analizado previamente. Dicho plano viene determinado para un valor concreto de θ_0 , el grado de libertad asociado a la articulación de la cintura. Como en el caso del planar, los C-obstáculos forman una familia de curvas parametrizadas por un parámetro. De hecho, es exactamente la misma familia de curvas, ya que el efecto de θ_0 es seleccionar el plano donde se produce la interacción con el obstáculo. En tres dimensiones la forma del obstáculo transformado depende únicamente del radio r donde se encuentra. Dependiendo de la posición del punto, esta primitiva se traslada en el C-espacio una cantidad ϕ en θ_1 y ϕ_0 en θ_0 , donde ϕ y ϕ_0 son las variables angulares del obstáculo, considerando coordenadas esféricas.

El método propuesto por [?] para robots planares y obstáculos poligonales se generaliza en un trabajo posterior [?] para robots espaciales y obstáculos poliédricos. En [?] se presenta una formulación algebraica de los límites de los C-obstáculos para robots espaciales con desacoplo cinemático. Esta formulación se aplica a un robot PUMA para obtener las ecuaciones parametrizadas del límite de los C-obstáculos. Sin embargo, en este método se supone una fuerte simplificación, como es la de modelar el brazo por una cadena de dos articulaciones esféricas. Además, solamente considera la parte del límite del C-obstáculo debida a choques con el efector final.

Existe otro trabajo [?] para obtener las ecuaciones del límite de los C-

obstáculos para manipuladores que trabajan en entornos poliédricos. Los obstáculos se modelan como uniones de caras triangulares, y los manipuladores mediante segmentos de línea. Las ecuaciones del límite de los C-obstáculos se calculan a partir de las condiciones que definen la intersección entre las caras triangulares y los segmentos de línea. Además, se muestra que las expresiones explícitas de los límites se pueden obtener para manipuladores cuya cinemática se represente con la notación de Denavit-Hatenberg. Más concretamente, se muestra que la ecuación del límite para la variable de una determinada articulación se puede resolver explícitamente en función de las variables de las articulaciones previas. Para validar el método, se aplica a un robot Adept cuyos elementos se aproximan por una unión de cilindros. Los obstáculos se aumentan en todas las direcciones por el radio de los cilindros y los elementos del robot se reducen a los ejes del cilindro. Se analizan las intersecciones en los ejes del cilindro y las uniones de los triángulos que aproximan a las superficies de las obstáculos aumentados.

3.2.3. Consideraciones finales

Una vez presentados de forma detallada los métodos más destacados para representar los obstáculos en el C-espacio de manipuladores articulados, se comentarán los inconvenientes más importantes que cada grupo, en conjunto, posee.

1. En los métodos que se engloban en el primer grupo es necesario construir una representación, más o menos aproximada, de los elementos que constituyen el robot para un conjunto elevado de configuraciones. En alguno de ellos hay que construirla para todas las configuraciones. Esto supone una elevada carga computacional asociada al cálculo de la cinemática directa. Además, habría que aplicar de forma explícita un test de colisiones para todos los elementos en todas las configuraciones consideradas. Este test de colisiones sería realizado con los métodos expuestos para los robots móviles.
2. Respecto al segundo grupo, hay que destacar que no existen muchos trabajos dedicados a robots articulados, y concretamente a brazos espaciales.

Los procedimientos propuestos calculan explícitamente las ecuaciones del límite de los C-obstáculos realizando una serie de simplificaciones. En primer lugar, simplifican la geometría de los elementos del robot y de los obstáculos. Concretamente, modelan los objetos mediante regiones algebraicas. Esto supone aproximar tanto a los elementos del robot como a los obstáculos por un conjunto de formas geométricas. Incluso, se llega a modelar al robot mediante segmentos de línea. Además, existen distintas técnicas que sustituyen la estructura mecánica del robot por otra más fácil de analizar. Todas estas simplificaciones, en general, son muy restrictivas para manipuladores articuladores trabajando en espacios tridimensionales, aunque no tanto para planares.

En segundo lugar, no todas las técnicas tienen en cuenta las colisiones entre todos los elementos que constituyen el brazo y los obstáculos. Normalmente, se restringen al estudio de los contactos con el efector final del robot.

Como consecuencias de estas simplificaciones, aunque con estos métodos se logre disminuir el tiempo de computación, los C-obstáculos que se obtienen son una versión aproximada de los reales.

Capítulo 4

Formalismo Matemático

Propuesto

En este capítulo se va presentar el formalismo matemático que se ha planteado para proyectar los obstáculos en el espacio de las configuraciones. En concreto, se propone una nueva representación de los C-obstáculos basada en la definición de tres funciones, que representan respectivamente: al robot en una determinada configuración, a los obstáculos en el espacio de trabajo y a los obstáculos en el espacio de las configuraciones. Esta última función, que toma valores sobre el espacio de las configuraciones, se define por medio de una integral sobre el espacio de trabajo del producto de las otras dos funciones. Así, evaluando si dicha función es definida positiva, se decide si una configuración pertenece a la región de los C-obstáculos.

Una vez que en la primera sección se ha planteado la base teórica propuesta, en la siguiente se muestra cómo se puede simplificar el cálculo de esa función, si se cumplen unas determinadas condiciones. Esa expresión simplificada conduce a proponer una nueva donde aparece el producto de convolución de las funciones que definen al robot y a los obstáculos sobre un determinado conjunto de variables, que depende de las estructuras robóticas concretas. Este es el objetivo de la siguiente sección. Para finalizar este nuevo planteamiento se propone aplicar el Teorema de Convolución para el cálculo de dicha integral.

En capítulos sucesivos se aplicará este formalismo a las estructuras robóticas

más representativas. Entre ellas cabe destacar los robots móviles con cualquier forma geométrica y los articulados más típicos, como son: el PUMA, el SCARA y el de Stanford. Se obtendrán distintos algoritmos para calcular los C-obstáculos que pondrán de manifiesto las ventajas del procedimiento respecto de los revisados en el capítulo previo.

4.1. Base Teórica Propuesta

En esta sección se va a proponer una nueva representación de los C-obstáculos, basada en la integral del producto de dos funciones: una que representa al robot y la otra a los obstáculos en el espacio de trabajo. Se comprobará que esta definición coincide con la propuesta en [?].

Sean W y C el espacio de trabajo y el espacio de las configuraciones de un robot, respectivamente. Así, la función A que se propone para representar al robot situado en un punto $x \in W$ en la configuración $q \in C$ vendría dada por la siguiente definición.

Definición 9 Sea $\mathbf{A}(q)$ el subconjunto de W que representa al robot en la configuración q . Se define la función $A : C \times W \rightarrow R$ como

$$A(q, x) = \begin{cases} 1 & \text{si } x \in \mathbf{A}(q) \\ 0 & \text{si } x \notin \mathbf{A}(q) \end{cases} \quad (4.1)$$

De la misma forma, la función B que se propone para representar a los obstáculos en el espacio de trabajo, estaría definida como se muestra a continuación.

Definición 10 Sea \mathbf{B} el subconjunto de W formado por los obstáculos. Se define la función $B : W \rightarrow R$ como

$$B(x) = \begin{cases} 1 & \text{si } x \in \mathbf{B} \\ 0 & \text{si } x \notin \mathbf{B} \end{cases} \quad (4.2)$$

Con estas dos funciones A y B , la nueva definición que se propone para calcular los C-obstáculos sería la siguiente:

Definición 11 Sea la función $CB : C \rightarrow R$ definida como

$$CB(q) = \int A(q, x)B(x)dx \quad \forall q \in C, \quad \forall x \in W \quad (4.3)$$

Se define la región \mathbf{CB}_f como el subconjunto de C que verifica

$$\mathbf{CB}_f = \{q \in C / CB(q) > 0\} \quad (4.4)$$

En el siguiente teorema se va a demostrar que la nueva definición coincide con la propuesta por Lozano-Pérez en [?], que es ampliamente aceptada y utilizada.

Teorema 2 Sea $\mathbf{CB} = \{q \in C / \mathbf{A}(q) \cap \mathbf{B} \neq \emptyset\}$ la región del C -obstáculo propuesta por [?]. Entonces, se cumple que $\mathbf{CB} = \mathbf{CB}_f$.

Demostración

a) $\mathbf{CB} \subset \mathbf{CB}_f$

Esto es equivalente a demostrar que

$$\forall q \in C \quad \text{si } q \in \mathbf{CB} \Rightarrow q \in \mathbf{CB}_f$$

es decir, que

$$\forall q \in C \quad \text{si } q \in \mathbf{CB} \Rightarrow CB(q) > 0$$

Se va a demostrar por reducción al absurdo. Supongamos que

$$\exists q \in \mathbf{CB} \quad \text{tal que } CB(q) = 0$$

Si $CB(q) = 0$, por la definición ?? se tiene que

$$\int A(q, x)B(x)dx = 0$$

y como, además, las funciones $A(q, x)$ y $B(x)$ son definidas positivas

$$\forall x \in W \quad A(q, x) \geq 0 \quad \text{y} \quad B(x) \geq 0$$

entonces tiene que cumplirse que

$$\forall x \in W \quad A(q, x)B(x) = 0$$

esto implica que

$$\forall x \in W \quad A(q, x) = 0 \quad \text{ó} \quad B(x) = 0$$

Con lo que se tiene que

$$\forall x \in W \quad x \notin \mathbf{A}(q) \quad \text{ó} \quad x \notin \mathbf{B}$$

y, por tanto

$$\forall x \in W \quad x \notin \mathbf{A}(q) \cap \mathbf{B}$$

Como consecuencia $q \notin \mathbf{CB}$, llegando a una contradicción con la hipótesis establecida.

b) $\mathbf{CB}_f \subset \mathbf{CB}$

Esta condición equivale a demostrar que

$$\forall q \in \mathcal{C} \quad \text{si } q \in \mathbf{CB}_f \Rightarrow q \in \mathbf{CB}$$

es decir, que

$$\forall q \in \mathcal{C} \quad \text{si } q \in \mathbf{CB}_f \Rightarrow \mathbf{A}(q) \cap \mathbf{B} \neq \emptyset$$

Esto se va a demostrar por reducción al absurdo. Supongamos que

$$\exists q \in \mathbf{CB}_f \quad \text{tal que } \mathbf{A}(q) \cap \mathbf{B} = \emptyset$$

Esto implica que

$$\forall x \in W \quad x \notin \mathbf{A}(q) \cap \mathbf{B}$$

y, por tanto, que

$$\forall x \in W \quad x \notin \mathbf{A}(q) \quad \text{ó } x \notin \mathbf{B}$$

Entonces, por las definiciones ?? y ??, se tiene que

$$\forall x \in W \quad A(q, x) = 0 \quad \text{ó } B(x) = 0$$

Esto lleva a que

$$\forall x \in W \quad A(q, x)B(x) = 0$$

y, por la definición ??,

$$CB(q) = 0$$

Como consecuencia $q \notin \mathbf{CB}_f$, que está en contradicción con la hipótesis inicial.

Corolario 1 *Sea \mathbf{C}_{free} el conjunto de configuraciones libres de colisiones. Entonces,*

$$q \in \mathbf{C}_{\text{free}} \Leftrightarrow CB(q) = 0$$

Luego, para conocer si el robot en una configuración dada q colisiona o no con los obstáculos es necesario calcular la función $CB(q)$ (??). En este cálculo juega un papel fundamental la función $A(q, x)$ (??).

4.2. Propiedades de los C-obstáculos

A continuación se van a presentar tres propiedades sobre la representación de los obstáculos en el C-espacio. Las dos primeras son válidas para cualquier tipo de robots, mientras que la tercera está relacionada exclusivamente con robots articulados. Con la definición de C-obstáculo propuesta (??) se pueden demostrar, de forma inmediata.

- Propiedad 1

Si \mathbf{CB}_f^1 y \mathbf{CB}_f^2 son, respectivamente, las regiones de los C-obstáculos donde se proyectan los obstáculos \mathbf{B}^1 y \mathbf{B}^2 , entonces la región \mathbf{CB}_f^{12} donde se proyecta $\mathbf{B}^1 \cup \mathbf{B}^2$ se obtiene como

$$\mathbf{CB}_f^{12} = \mathbf{CB}_f^1 \cup \mathbf{CB}_f^2$$

Esta propiedad se ha utilizado en muchos trabajos ([?] y [?]) para encontrar los C-obstáculos debidos a objetos complejos si se conocen los correspondientes a formas de obstáculos elementales en los que se pueden descomponer o aproximar.

- Propiedad 2

Si \mathbf{CB}_f^1 y \mathbf{CB}_f^2 son, respectivamente, las regiones de los C-obstáculos donde se proyectan los obstáculos \mathbf{B}^1 y \mathbf{B}^2 , y $\mathbf{B}^1 \subseteq \mathbf{B}^2$ entonces

$$\mathbf{CB}_f^2 = \mathbf{CB}_f^1 \cup \mathbf{CB}_f^2$$

o bien

$$\mathbf{CB}_f^1 \subseteq \mathbf{CB}_f^2$$

Esta propiedad puede reducir el tiempo de cálculo asociado con la ejecución de un determinado algoritmo diseñado para la representación de los C-obstáculos. Si únicamente se proyecta el contorno de un obstáculo al C-espacio entonces se tiene el límite del C-obstáculo. Esta idea se ha utilizado en algunos procedimientos de planificación ([?] y [?]), que trabajan con C-espacios discretos, para calcular el C-obstáculo. En primer lugar, calculan su límite con un determinado algoritmo y después utilizan alguna técnica para rellenarlo que, en general, proviene de Informática Gráfica. Además de reducir el tiempo de cálculo dedicado al cálculo del C-obstáculo, se reduce el tiempo empleado en la detección de colisiones, ya que solamente es necesario realizarla con el límite y,

así, se tiene garantizada la protección con el interior. También es la base de los trabajos que obtienen los contornos de los C-obstáculos.

- Propiedad 3

Sea \mathbf{A} un robot articulado formado por dos elementos \mathbf{A}_1 y \mathbf{A}_2 . Si \mathbf{CB}_{1f} y \mathbf{CB}_{2f} son, respectivamente, las regiones de los C-obstáculos debidas a los elementos \mathbf{A}_1 y \mathbf{A}_2 donde se proyecta el obstáculo \mathbf{B} , entonces la región \mathbf{CB}_f del C-obstáculo debida al robot \mathbf{A} donde se proyecta \mathbf{B} se obtiene como

$$\mathbf{CB}_f = \mathbf{CB}_{1f} \cup \mathbf{CB}_{2f} \quad (4.5)$$

Esta propiedad es la base del cálculo de los C-obstáculos para robots formados por varios elementos conectados mediante articulaciones prismáticas o de revolución. Se cumple gracias a la estructura mecánica de los manipuladores articulados. En la expresión ??, no se han considerado las colisiones entre los distintos elementos que integran la cadena cinemática del brazo, dado que este cálculo sólo es necesario efectuarlo un única vez para cada robot articulado. Por tanto, el tiempo de cálculo está particularmente asociado con determinar los dos términos de la expresión ??.

4.3. Parametrización de W y C

En la sección previa se presentaban las expresiones para las funciones B , CB y A , definidas, respectivamente, sobre los espacios W , C y $C \times W$, sin suponer ninguna parametrización específica de W y de C . En esta sección se va a seleccionar una representación tanto de C como de W en forma de una lista de parámetros reales, que permitirá obtener la correspondiente forma parametrizada de las funciones previamente definidas. Esto servirá de punto de partida para exponer, en la sección siguiente, un método que permite simplificar el cálculo de la función CB .

Una representación de W y de C viene dada seleccionando un sistema de referencia F_W para el espacio de trabajo y otro F_A para el robot. Respecto de ellos, se eligen las coordenadas de trabajo para C y W , de tal forma que un punto x de W vendrá expresado respecto del sistema de referencia F_W por las coordenadas

$$(x_1, x_2, \dots, x_n)$$

donde n es la dimensión de W . Una configuración q viene representada por una lista de parámetros

$$(q_1, q_2, \dots, q_m)$$

que especifican la posición y orientación de F_A con respecto a F_W , siendo m la dimensión de C . En general, se cumple que $n \leq m$.

Con esta parametrización de C y W la función A dada previamente (??)

$$\begin{aligned} A : C \times W &\longrightarrow R \\ (q_1, \dots, q_m) \times (x_1, \dots, x_n) &\longrightarrow A(q_1, \dots, q_m, x_1, \dots, x_n) \end{aligned}$$

estaría definida como

$$A(q_1, \dots, q_m, x_1, \dots, x_n) = \begin{cases} 1 & \text{si } (x_1, \dots, x_n) \in \mathbf{A}(q_1, \dots, q_m) \\ 0 & \text{si } (x_1, \dots, x_n) \notin \mathbf{A}(q_1, \dots, q_m) \end{cases} \quad (4.6)$$

donde $\mathbf{A}(q_1, \dots, q_m)$ es el subconjunto de W que representa al robot en la configuración (q_1, \dots, q_m) .

De la misma forma la función B , (??),

$$\begin{aligned} B : W &\longrightarrow R \\ (x_1, \dots, x_n) &\longrightarrow B(x_1, \dots, x_n) \end{aligned}$$

se definiría como

$$B(x_1, \dots, x_n) = \begin{cases} 1 & \text{si } (x_1, \dots, x_n) \in \mathbf{B} \\ 0 & \text{si } (x_1, \dots, x_n) \notin \mathbf{B} \end{cases} \quad (4.7)$$

siendo \mathbf{B} el subconjunto de W formado por los obstáculos.

La función CB , (??)

$$\begin{aligned} CB : C &\longrightarrow R \\ (q_1, \dots, q_m) &\longrightarrow CB(q_1, \dots, q_m) \end{aligned}$$

estaría definida como

$$CB(q_1, \dots, q_m) = \int A(q_1, \dots, q_m, x_1, \dots, x_n) B(x_1, \dots, x_n) dx_1 \dots dx_n \quad (4.8)$$

A continuación se va a comprobar cómo eligiendo unos sistemas de referencia y unas coordenadas adecuadas, tanto en el espacio de trabajo como en espacio de las configuraciones, se puede evaluar la integral (??) de forma más simple, reduciendo, por tanto, la carga computacional.

4.4. Simplificación del Cálculo de CB

La dificultad más importante para el cálculo de esta integral se encuentra en la evaluación de $A(q_1, \dots, q_m, x_1, \dots, x_n)$. Una forma de simplificar el cálculo es encontrar una función A' tal que la integral anterior (??) esté dada por

$$\int A'(0, \dots, 0, q_{r+1}, \dots, q_m, x_1 - q_1, \dots, x_r - q_r, x_{r+1}, \dots, x_n) B(x_1, \dots, x_n) dx_1 \cdots dx_n \quad (4.9)$$

De esta forma existe una dependencia funcional más sencilla en la función A' .

El problema es determinar si existe la función A' y cuál es su definición, lo que constituye el propósito del siguiente planteamiento. En primer lugar es necesario seleccionar los sistemas de referencia, F_A y F_W , para el robot y el espacio de trabajo, y después hay que elegir las coordenadas adecuadas para C y W . Esta determinación de sistemas de referencia y coordenadas se ha de realizar con el objetivo de encontrar un morfismo

$$\begin{aligned} \phi : C \times W &\longrightarrow C' \times W' \\ (q_1, \dots, q_m) \times (x_1, \dots, x_n) &\longrightarrow (q'_1, \dots, q'_m) \times (x'_1, \dots, x'_n) \end{aligned}$$

definido como

$$\begin{aligned} \forall j \in \{1, \dots, r\} & \quad x'_j = x_j - q_j \quad q'_j = 0 \\ \forall i \in \{r+1, \dots, m\} & \quad q'_i = q_i \\ \forall k \in \{r+1, \dots, n\} & \quad x'_k = x_k \end{aligned}$$

donde $1 \leq r \leq n$, de forma que la función $A = A' \circ \phi$ definida de $C' \times W'$ en R verifique que

$$\frac{\partial A'}{\partial q'_j} = 0 \quad \forall j \in \{1 \dots r\} \quad (4.10)$$

De este modo se consigue que A' sea independiente de un subconjunto de q , es decir,

$$A'(0, \dots, 0, q'_{r+1}, \dots, q'_m, x'_1, \dots, x'_n)$$

o con la definición del morfismo ϕ

$$A'(0, \dots, 0, q_{r+1}, \dots, q_m, x_1 - q_1, \dots, x_r - q_r, x_{r+1}, \dots, x_n)$$

Por tanto, si es posible seleccionar dichos sistemas de referencia y coordenadas el cálculo de $CB(q)$ (??) vendría dado por

$$CB(q_1, \dots, q_m) = \int A(q_1, \dots, q_m, x_1, \dots, x_n) B(x_1, \dots, x_n) dx_1 \cdots dx_n$$

$$= \int (A' \circ \phi)(q_1, \dots, q_m, x_1, \dots, x_n) B(x_1, \dots, x_n) dx_1 \cdots dx_n$$

En definitiva, se obtiene la expresión (??) para el cálculo de $CB(q)$.

4.5. Cálculo de CB Utilizando la Convolución

Como se analizará a continuación, en el cálculo de $CB(q)$ aparece el producto de convolución de dos funciones sobre un conjunto de variables que depende de la estructura del robot considerado. Este aspecto será crucial a la hora de reducir la complejidad computacional, ya que para el cálculo de la convolución existen algoritmos fuertemente optimizados.

La convolución de dos funciones f_1 y f_2 definidas en R^n es una nueva función dada por

$$(f_1 * f_2)(x_1, \dots, x_n) = \int f_1(x'_1, \dots, x'_n) f_2(x_1 - x'_1, \dots, x_n - x'_n) dx'_1 \cdots dx'_n$$

Si para esas dos funciones f_1 y f_2 solamente se produjera convolución sobre algunas variables, digamos r con $1 \leq r \leq n$, se tendría

$$(f_1 * f_2)_{(x'_1, \dots, x'_r)} = \int f_1(x'_1, \dots, x'_n) f_2(x_1 - x'_1, \dots, x_r - x'_r, x_{r+1}, \dots, x_n) dx'_1 \cdots dx'_r$$

Teniendo en cuenta esta última definición, el producto de convolución aparece en el cálculo simplificado de CB (??) si previamente se realiza el siguiente cambio

$$\begin{aligned} \bar{A}'(0, \dots, 0, q_{r+1}, \dots, q_m, q_1 - x_1, \dots, q_r - x_r, x_{r+1}, \dots, x_n) &= \quad (4.11) \\ A'(0, \dots, 0, q_{r+1}, \dots, q_m, x_1 - q_1, \dots, x_r - q_r, x_{r+1}, \dots, x_n) \end{aligned}$$

Así, se obtiene la siguiente expresión para $CB(q)$

$$\int \bar{A}'(0, \dots, 0, q_{r+1}, \dots, q_m, q_1 - x_1, \dots, q_r - x_r, x_{r+1}, \dots, x_n) B(x_1, \dots, x_n) dx_1 \cdots dx_n \quad (4.12)$$

Entonces, en el cálculo de $CB(q)$, aparece el producto de convolución de las funciones \bar{A}' y B sobre el conjunto de variables $\{x_1, \dots, x_r\}$. Es decir, se llegaría a la siguiente expresión para $CB(q_1, \dots, q_m)$

$$\int (\bar{A}'_{(0, \dots, 0, q_{r+1}, \dots, q_m)} * B)_{(x_1, \dots, x_r)}(q_1, \dots, q_r, x_{r+1}, \dots, x_n) dx_{r+1} \cdots dx_n \quad (4.13)$$

Para llegar a esta expresión es fundamental la selección de los sistemas de referencia y coordenadas. Esta selección depende de la estructura mecánica del robot que se considere en cada caso.

4.6. Cálculo de CB Utilizando la Transformada de Fourier

En este apartado se mostrará cómo el cálculo de $CB(q)$ se puede realizar aplicando el Teorema de Convulsión. El objetivo no es otro que el de utilizar algoritmos fuertemente optimizados, ya existentes, que permitan reducir la complejidad computacional de este cálculo.

El Teorema de Convulsión establece que, dadas dos funciones $f_1(x)$ y $f_2(x)$ definidas en R cuyas Transformadas de Fourier (FT) son respectivamente $\mathcal{F}[f_1(x)]$ y $\mathcal{F}[f_2(x)]$ entonces es:

$$\mathcal{F}[(f_1 * f_2)(x)] = \mathcal{F}[f_1(x)] \cdot \mathcal{F}[f_2(x)]$$

Este teorema es muy útil, pues permite calcular el producto de convulsión de dos funciones como la Transformada de Fourier inversa del producto de las Transformadas de ambas funciones.

Considerando Transformadas en la expresión ?? del cálculo de CB y utilizando este teorema se llega a

$$\mathcal{F}[CB] = \int \mathcal{F} [\bar{A}'_{(0, \dots, 0, q_{r+1}, \dots, q_m)}(q_1, \dots, q_r, x_{r+1}, \dots, x_n)]_{(x_1, \dots, x_r)} \cdot \mathcal{F} [B(q_1, \dots, q_r, x_{r+1}, \dots, x_n)]_{(x_1, \dots, x_r)} dx_{r+1} \dots dx_n$$

Los subíndices (x_1, \dots, x_r) indican que se trata de Transformadas de dimensión r de funciones definidas en R^n . Sobre el resto de las coordenadas (x_{r+1}, \dots, x_n) donde no se produce convulsión se efectúa la integración.

Dependiendo de la estructura mecánica de cada robot en particular, esta expresión tendrá una forma concreta. Sin embargo, se puede observar que, a diferencia de otros métodos, la complejidad computacional no depende de la forma ni del número de vértices del robot y de los obstáculos. Únicamente depende de la rapidez con la que se realice el producto de convulsión o la Transformada de Fourier. Existen algoritmos muy optimizados que reducen drásticamente la carga computacional si se utiliza la Transformada de Fourier. Esta característica será analizada profundamente cuando se presente la expresión discreta y los algoritmos correspondientes para la construcción de los C-obstáculos.

Capítulo 5

Aplicación del Formalismo para Robots Móviles

En el capítulo previo se ha propuesto, de forma general, un formalismo matemático para calcular los C-obstáculos. En el capítulo actual y en el siguiente dicho formalismo se particularizará tanto a robots móviles como a articulados. Se ha comenzado por la aplicación a robots móviles porque en ellos se puede intuir de forma fácil que los C-obstáculos se pueden obtener calculando el producto de convolución del robot con los obstáculos. De forma gráfica la convolución de dos funciones se representa considerando el movimiento libre de un objeto definido por una de las funciones sobre un espacio donde existe otro objeto definido por otra. Asimismo, un C-obstáculo se puede obtener desplazando el robot móvil a lo largo del contorno del obstáculo en el espacio de trabajo. Sin embargo, cada caso particular que se va a tratar presenta diversas peculiaridades.

En este capítulo se va a aplicar el formalismo a tres robots móviles diferentes. En las tres situaciones, el robot se mueve por una superficie plana. En primer lugar se ha elegido un robot con forma de disco sin marcas, un ejemplo sencillo que facilitará la comprensión del método y del formalismo matemático general propuesto en el capítulo anterior. Posteriormente, se analizará el caso de una plataforma móvil con cualquier forma geométrica que puede desplazarse y girar libremente en R^2 . Esto permitirá extender, finalmente, el procedimiento para el cálculo del C-espacio de una plataforma con tres dimensiones que se mueve

sobre un plano.

5.1. Robot Circular Móvil en el Plano

En esta sección, se considera un robot \mathbf{A} con forma de disco sin ninguna marca, con simetría axial, que se desplaza y gira libremente en un espacio de trabajo $W \subset \mathbb{R}^2$.

En W se define un sistema de referencia F_W fijo. En este caso es adecuado trabajar en W con coordenadas cartesianas. Además, se define un sistema de referencia F_A que se mueve con el robot. Debido a la simetría axial del robot se sitúa el origen de F_A en el centro del robot. En la gráfica ?? se muestra la elección de los sistemas de referencia que se ha efectuado .

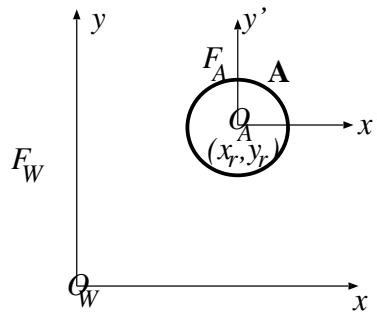


Figura 5.1: Sistemas de referencia para robot circular móvil en el plano

Dada su simetría, una configuración q de \mathbf{A} vendría dada únicamente por la posición de F_A con respecto a F_W , a la que denotamos, por (x_r, y_r) . Por tanto, el espacio de las configuraciones se puede representar como un espacio euclídeo $C \subset \mathbb{R}^2$.

Entonces la función B definida sobre W en \mathbb{R} estaría dada por

$$B(x, y) = \begin{cases} 1 & \text{si } (x, y) \in \mathbf{B} \\ 0 & \text{si } (x, y) \notin \mathbf{B} \end{cases} \quad (5.1)$$

donde \mathbf{B} es el subconjunto de W que representa a los obstáculos.

En cuanto a la función definida en la expresión ?? $A : C \times W \rightarrow \mathbb{R}$ estaría

dada por

$$A(x_r, y_r, x, y) = \begin{cases} 1 & \text{si } (x, y) \in \mathbf{A}_{(x_r, y_r)} \\ 0 & \text{si } (x, y) \notin \mathbf{A}_{(x_r, y_r)} \end{cases} \quad (5.2)$$

donde $\mathbf{A}_{(x_r, y_r)}$ es el subconjunto de puntos de W que representa al robot en la configuración (x_r, y_r) .

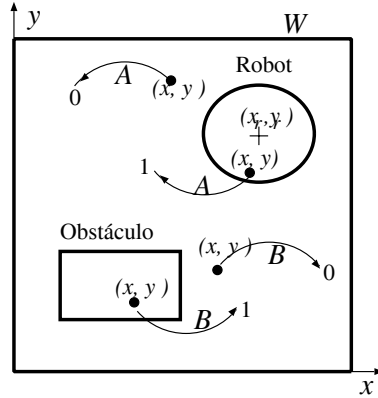


Figura 5.2: Definición de A y B para robot circular móvil

La definición de estas funciones se puede observar en la figura ???. Con ellas, la función $CB : C \rightarrow R$ (??) estaría definida como

$$CB(x_r, y_r) = \int A(x_r, y_r, x, y)B(x, y)dx dy \quad (5.3)$$

Como se ha puesto de manifiesto en el formalismo matemático propuesto, para simplificar el cálculo de esta integral hay que encontrar una función A' definida sobre $C' \times W'$ en R , de forma que sea independiente de un subconjunto de parámetros que definen una configuración. Para ello, hay que elegir unos nuevos sistemas de referencia. Concretamente, se han elegido $F_{W'} \equiv F_A$ y $F_{A'} \equiv F_A$ (figura ??).

Respecto de estos sistemas, C' contiene una única configuración dada por $q = (0, 0)$, con lo que la función A'

$$\begin{aligned} A' : C' \times W' &\longrightarrow R \\ 0, 0, x', y' &\longrightarrow A'(0, 0, x', y') \end{aligned}$$

estaría dada por

$$A'(0, 0, x', y') = \begin{cases} 1 & \text{si } (x', y') \in \mathbf{A}_{(0,0)} \\ 0 & \text{si } (x', y') \notin \mathbf{A}_{(0,0)} \end{cases} \quad (5.4)$$

donde $\mathbf{A}_{(0,0)}$ es el subconjunto de puntos que representa al robot en la configuración $q = (0, 0)$ respecto de $F_{W'}$ (figura ??).

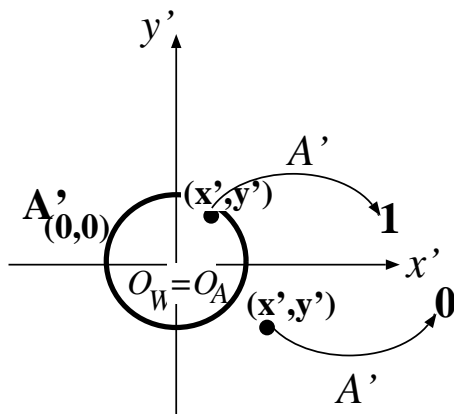


Figura 5.3: Definición de A' para robot circular móvil

Solamente es necesario definir un morfismo ϕ tal que $A = A' \circ \phi$. Este morfismo produce el cambio de coordenadas de $C \times W$ en $C' \times W'$. Su definición sería

$$\begin{aligned} x' &= x - x_r & y' &= y - y_r \\ x'_r &= 0 & y'_r &= 0 \end{aligned}$$

y con ello

$$A(x_r, y_r, x, y) = (A' \circ \phi)(x_r, y_r, x, y) = A'(0, 0, x - x_r, y - y_r)$$

Dada la definición de las funciones A (??) y A' (??), se tiene que

$$A'(0, 0, x - x_r, y - y_r) = A(0, 0, x - x_r, y - y_r)$$

y, por tanto, se llega a que

$$A(x_r, y_r, x, y) = A(0, 0, x - x_r, y - y_r) \quad (5.5)$$

Así se obtiene que la función característica A que define al robot en una configuración cualquiera (x_r, y_r) y en un punto del espacio de trabajo (x, y) coincide con dicha función, considerando al robot situado en la configuración $(0, 0)$ y efectuando una traslación en las coordenadas del espacio de trabajo. Por similitud con la notación del robot en una configuración dada, se denota por

$$A_{(0,0)}(x - x_r, y - y_r) = A(0, 0, x - x_r, y - y_r) \quad (5.6)$$

A partir de (??) y (??) se obtiene

$$A(x_r, y_r, x, y) = A_{(0,0)}(x - x_r, y - y_r) \quad (5.7)$$

Con las expresiones (??) y (??), se puede calcular CB como

$$\begin{aligned} CB(x_r, y_r) &= \int A(x_r, y_r, x, y)B(x, y)dxdy \\ &= \int A_{(0,0)}(x - x_r, y - y_r)B(x, y)dxdy \end{aligned} \quad (5.8)$$

De esta forma, se observa que pueden encontrarse unas coordenadas para el espacio de trabajo, directamente relacionadas con la estructura física del robot, y una representación (x_r, y_r) de C , que permiten trabajar en un determinado atlas. En éste, existe un morfismo que, con funciones características del robot, hacen al robot independiente de alguna de las coordenadas del espacio de las configuraciones. Se consigue simplificar, así, el cálculo de $CB(q)$.

Si en la expresión de CB (??) se considera la función

$$\bar{A}_{(0,0)}(x_r - x, y_r - y) = A_{(0,0)}(x - x_r, y - y_r) \quad (5.9)$$

entonces aparece el producto de convolución de dos funciones definidas en R^2 sobre las dos variables

$$CB(x_r, y_r) = (\bar{A}_{(0,0)} * B)(x_r, y_r) \quad (5.10)$$

La función $\bar{A}_q(-x)$ coincide con $A_q(x)$ y representa al robot en la configuración q con sus coordenadas invertidas.

Tomando transformadas de Fourier en (??) y utilizando el Teorema de Convolución se llega a

$$\mathcal{F}[CB(x_r, y_r)] = \mathcal{F}[\bar{A}_{(0,0)}(x_r, y_r)] \mathcal{F}[B(x_r, y_r)] \quad (5.11)$$

En ella se observa que la transformada de Fourier de CB es el producto de transformadas, en dos dimensiones, de dos funciones definidas en R^2 : \bar{A} que define al robot y B que define a los obstáculos. El cálculo de los obstáculos en el espacio de las configuraciones se obtendría sin más que calcular la transformada inversa de la expresión (??).

5.2. Plataforma 2D Móvil en el Plano

En este apartado, se va a considerar que el robot es una plataforma de dos dimensiones con cualquier forma geométrica: rectangular, cuadrada, en estrella, o cualquier otra que uno pueda imaginar, que se mueve sobre un plano. En particular se supone que es ovalada, aunque el planteamiento no sufre ninguna modificación si la forma es otra. Inicialmente se considera que el robot se desplaza con una orientación fija. Posteriormente se trata el caso de una plataforma que se desplaza y gira libremente en el plano.

5.2.1. Plataforma móvil con orientación fija

En este caso se supone que la plataforma se desplaza libremente con una orientación fija en un espacio de trabajo $W \subset R^2$ ocupado por obstáculos.

Del mismo modo que para el robot con forma de disco, se define en W un sistema de coordenadas cartesianas F_W fijo dado por (x, y) , y otro sistema de coordenadas cartesianas F_A móvil rígidamente unido a la plataforma (figura ??), cuyo origen se encuentra en cualquier punto (x_r, y_r) del robot.

El análisis es similar al realizado en la sección previa, donde una configuración q del robot vendría dada por la posición (x_r, y_r) de F_A con respecto a F_W . La expresión para el cálculo de CB coincide con la dada para el robot con forma de disco (??), sin más que modificar la expresión para $\mathbf{A}_{(0,0)}$, que representaba al robot tipo disco situado en el origen de coordenadas de F_W , por la de la plataforma móvil con una orientación fija situada también en el origen.

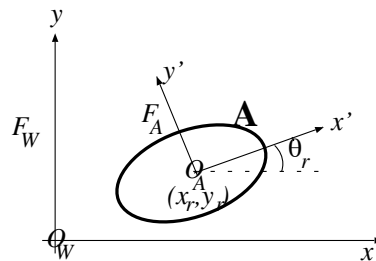


Figura 5.4: Sistemas de referencia para plataforma móvil en el plano

5.2.2. Plataforma móvil que se desplaza y gira libremente

A continuación se supone que la plataforma móvil puede trasladarse y rotar libremente en un espacio de trabajo $W \subset R^2$. Se considera que tiene forma de óvalo, al igual que el ejemplo previo, aunque cualquier otra forma geométrica puede ser considerada. En este caso sería necesario modificar en parte el análisis anteriormente efectuado.

Se realiza la misma elección de sistemas de referencia F_W y F_A que la efectuada previamente (figura ??).

La representación de C vendría dada ahora parametrizando cada configuración q por $(x_r, y_r, \theta_r) \in R^2 \times [0, 2\pi]$, donde (x_r, y_r) y θ_r son, respectivamente, el origen y la orientación de F_A con respecto a F_W . El espacio de las configuraciones sería de dimensión tres, $C \subset R^2 \times [0, 2\pi]$.

La definición de la función B (figura ??) coincide con la expresión (?). Sin embargo, la función $A : C \times W \rightarrow R$ estaría dada (figura ??) por

$$A(x_r, y_r, \theta_r, x, y) = \begin{cases} 1 & \text{si } (x, y) \in \mathbf{A}_{(x_r, y_r, \theta_r)} \\ 0 & \text{si } (x, y) \notin \mathbf{A}_{(x_r, y_r, \theta_r)} \end{cases} \quad (5.12)$$

donde $\mathbf{A}_{(x_r, y_r, \theta_r)}$ es el subconjunto de W que representa al robot en la configuración (x_r, y_r, θ_r) .

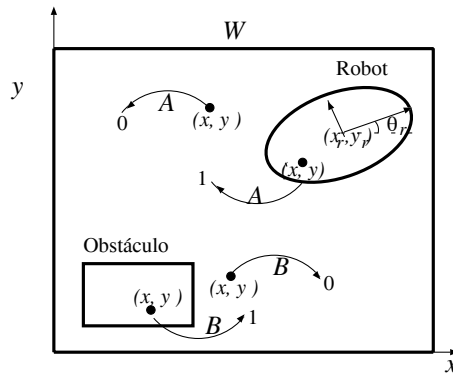


Figura 5.5: Definición de A y B para una plataforma móvil

Así $CB(q)$ estaría definida como

$$CB(x_r, y_r, \theta_r) = \int A(x_r, y_r, \theta_r, x, y)B(x, y)dx dy \quad (5.13)$$

Para simplificar esta expresión, hay que encontrar una función A' definida de $C' \times W'$ en R . Se eligen los nuevos sistemas de referencia (figura ??):

- El origen de $F_{W'}$ coincidente con el de F_A
- $F_{A'} \equiv F_A$

Respecto de estos sistemas, C' contiene un conjunto de configuraciones dado por $(0, 0, \theta'_r)$, siendo $\theta'_r = \theta_r$. Con esta elección de sistemas de referencia se tiene que $A' : C' \times W' \rightarrow R$ estaría definida como

$$A'(0, 0, \theta_r, x', y') = \begin{cases} 1 & \text{si } (x', y') \in \mathbf{A}_{(0,0,\theta_r)} \\ 0 & \text{si } (x', y') \notin \mathbf{A}_{(0,0,\theta_r)} \end{cases} \quad (5.14)$$

donde $\mathbf{A}_{(0,0,\theta_r)}$ es el subconjunto de puntos que representa al robot en la configuración $q = (0, 0, \theta_r)$ respecto de $F_{W'}$. Su definición, así como la del subconjunto, se puede observar en la figura ??.

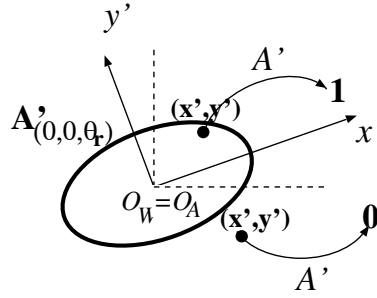


Figura 5.6: Definición de A' para la plataforma móvil

Como se tenía que $C \times W \xrightarrow{A} R$ y $C' \times W' \xrightarrow{A'} R$ sólo hay que definir un morfismo $C \times W \xrightarrow{\phi} C' \times W'$ que efectúe la traslación de coordenadas. La definición de este morfismo

$$\phi : C \times W \longrightarrow C' \times W' \quad x_r, y_r, \theta_r, x, y \longrightarrow x'_r, y'_r, \theta'_r, x', y'$$

donde

$$\begin{aligned} x' &= x - x_r & y' &= y - y_r \\ x'_r &= 0 & y'_r &= 0 & \theta'_r &= \theta_r \end{aligned}$$

Con ϕ se cumple que $A = A' \circ \phi$, entonces

$$A(x_r, y_r, \theta_r, x, y) = (A' \circ \phi)(x_r, y_r, \theta_r, x, y) = A'(0, 0, \theta_r, x - x_r, y - y_r)$$

Dada la definición de las funciones A (??) y A' (??), se tiene que

$$A'(0, 0, \theta_r, x - x_r, y - y_r) = A(0, 0, \theta_r, x - x_r, y - y_r)$$

y por tanto

$$A(x_r, y_r, \theta_r, x, y) = A(0, 0, \theta_r, x - x_r, y - y_r) \quad (5.15)$$

De esta forma se observa que la función que define el robot permanece invariante si se efectúa una traslación respecto de los ejes (x, y) con una orientación θ_r fija, pasando el robot de estar situado en una configuración (x_r, y_r, θ_r) a estar situado en $(0, 0, \theta_r)$. Por similitud con la notación del robot en una configuración dada, se denota por

$$A_{(0,0,\theta_r)}(x - x_r, y - y_r) = A(0, 0, \theta_r, x - x_r, y - y_r)$$

De esta forma se observa que se puede encontrar un sistema de coordenadas para el espacio de trabajo, directamente relacionado con la estructura física del robot, y una representación (x_r, y_r, θ_r) de C , que permiten trabajar en un determinado atlas. En éste, para cada carta existe un morfismo que, con funciones características del robot, hace al robot independiente de alguna de las coordenadas del espacio de las configuraciones. Así, con (??) y (??), se puede calcular CB de la siguiente forma

$$\begin{aligned} CB(x_r, y_r, \theta_r) &= \int A(x_r, y_r, \theta_r, x, y) B(x, y) dx dy \\ &= \int A_{(0,0,\theta_r)}(x - x_r, y - y_r) B(x, y) dx dy \quad (5.16) \end{aligned}$$

Para cada orientación fija θ_r de A se obtendría un $CB(x_r, y_r, \theta_r)$ idéntico al obtenido cuando el robot únicamente puede trasladarse en el plano. Variando θ_r en el intervalo angular particular se tienen los obstáculos en el C-espacio $R^2 \times S^1$, cuyas coordenadas son (x_r, y_r, θ_r) . Cada plano perpendicular a θ_r es el C-obstáculo de dos dimensiones (x_r, y_r) para la correspondiente orientación fija de la plataforma.

Con la definición del producto de convolución de dos funciones definidas en R^2 sobre las dos variables, la expresión de CB (??) sería

$$CB(x_r, y_r, \theta_r) = (\bar{A}_{(0,0,\theta_r)} * B)(x_r, y_r) \quad (5.17)$$

donde $\bar{A}_{(0,0,\theta_r)}(x_r - x, y_r - y) = A_{(0,0,\theta_r)}(x - x_r, y - y_r)$ se ha efectuado una inversión en las coordenadas. Aplicando el Teorema de Convulación sobre (??) se obtiene

$$\mathcal{F}[CB(x_r, y_r, \theta_r)] = \mathcal{F}[\bar{A}_{(0,0,\theta_r)}(x_r, y_r)] \mathcal{F}[B(x_r, y_r)] \quad (5.18)$$

La representación de los obstáculos en el C-espacio se obtiene calculando la transformada de Fourier inversa de la expresión (??). En dicha expresión aparece el producto de transformadas en dos dimensiones.

5.3. Plataforma 3D Móvil sobre el Plano

El tercer robot móvil que se va a estudiar es una plataforma 3D con cualquier forma geométrica que mueve sobre un plano. En concreto, se supone que tiene forma prismática. El planteamiento es válido si fuera cualquier otro poliedro, excepto un cilindro, cuya simetría axial reduce el problema considerablemente. El robot prismático puede rotar y desplazarse en un espacio de trabajo $W \subset R^3$, con la restricción de que el movimiento se efectúe sobre el plano $z = 0$.

En primer lugar, es necesario definir en W un sistema de coordenadas cartesianas F_W fijo dado por (x, y, z) , y otro sistema de coordenadas cartesianas F_A móvil, rigidamente unido al robot (figura ??), y cuyo origen se encuentra en uno de los vértices del prisma situado en el plano $z = 0$.

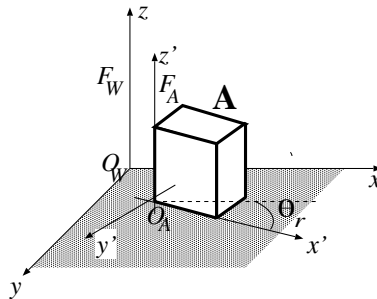


Figura 5.7: Sistemas de referencia para un prisma móvil sobre el plano

El problema es muy similar al planteado en la sección anterior, donde se tenía una plataforma 2D que podía desplazarse y girar libremente en el plano. El espacio de las configuraciones tendría tres dimensiones, $C \subset R^2 \times [0, 2\pi]$, al igual

que el caso previo. La representación de C también vendría dada parametrizando cada configuración q por (x_r, y_r, θ_r) , donde (x_r, y_r) y θ_r son, respectivamente, el origen y la orientación de F_A con respecto a F_W .

La función B definida sobre W en R estaría dada ahora por

$$B(x, y, z) = \begin{cases} 1 & \text{si } (x, y, z) \in \mathbf{B} \\ 0 & \text{si } (x, y, z) \notin \mathbf{B} \end{cases} \quad (5.19)$$

siendo \mathbf{B} es el subconjunto de W que representa a los obstáculos.

En cuanto a la función $A : C \times W \rightarrow R$ definida en (??), estaría dada, en este caso, por

$$A(x_r, y_r, \theta_r, x, y, z) = \begin{cases} 1 & \text{si } (x, y, z) \in \mathbf{A}_{(x_r, y_r, \theta_r)} \\ 0 & \text{si } (x, y, z) \notin \mathbf{A}_{(x_r, y_r, \theta_r)} \end{cases} \quad (5.20)$$

donde $\mathbf{A}_{(x_r, y_r, \theta_r)}$ el subconjunto de W que representa al robot en la configuración (x_r, y_r, θ_r) . La definición de estas funciones se puede observar en la figura ??.

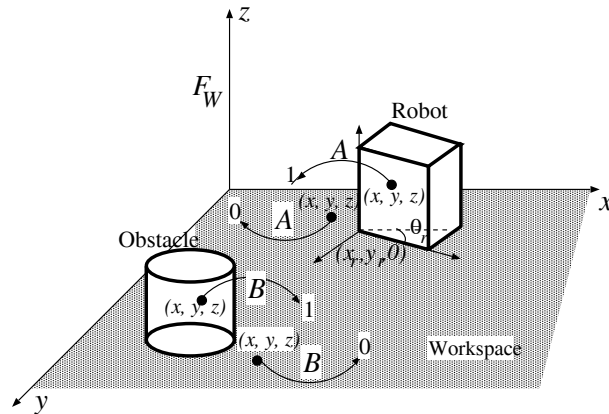


Figura 5.8: Definición de A y B para un prisma móvil

A partir de ambas funciones, el cálculo de CB vendría dado por la expresión

$$CB(x_r, y_r, \theta_r) = \int A(x_r, y_r, \theta_r, x, y, z)B(x, y, z)dx dy dz \quad (5.21)$$

Para simplificar este cálculo hay que buscar una función $A' : C' \times W' \rightarrow R$ que no dependa de un subconjunto de parámetros que definen una configuración. Para ello A' se define respecto de los sistemas de referencia $F_{W'}$ y $F_{A'}$. En este caso se han elegido de la siguiente forma (figura ??):

- El origen de $F_{W'}$ coincidente con el de F_A
- $F_{A'} \equiv F_A$

Así C' está formado por un conjunto de configuraciones $(0, 0, \theta'_r)$, siendo $\theta'_r = \theta_r$.

La función $A' : C' \times W' \rightarrow R$ estaría dada por

$$A'(0, 0, \theta_r, x', y', z') = \begin{cases} 1 & \text{si } (x', y', z') \in \mathbf{A}_{(0,0,\theta_r)} \\ 0 & \text{si } (x', y', z') \notin \mathbf{A}_{(0,0,\theta_r)} \end{cases} \quad (5.22)$$

donde $\mathbf{A}_{(0,0,\theta_r)}$ es el subconjunto de puntos que representa al robot en la configuración $q = (0, 0, \theta_r)$ respecto de $F_{W'}$. En la figura ?? aparece la definición de la función A' así como del subconjunto $\mathbf{A}_{(0,0,\theta_r)}$.

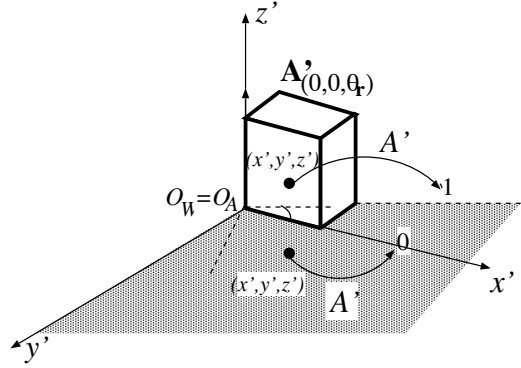


Figura 5.9: Definición de $A_{(0,0,\theta_r)}$ para el prisma móvil

Hay que definir un morfismo ϕ que realice el cambio de coordenadas de $C \times W$ a $C' \times W'$. En este caso es lógico definirlo como

$$\begin{aligned} \phi : C \times W &\longrightarrow C' \times W' \\ x_r, y_r, \theta_r, x, y, z &\longrightarrow x'_r, y'_r, \theta'_r, x', y', z' \end{aligned}$$

donde

$$\begin{aligned} x' &= x - x_r & y' &= y - y_r & z' &= z \\ x'_r &= 0 & y'_r &= 0 & \theta'_r &= \theta_r \end{aligned}$$

ya que se está efectuando una traslación del robot respecto de los ejes x, y manteniendo constante su orientación. Con ϕ se cumple que $A = (A' \circ \phi)$, y entonces se tiene

$$A(x_r, y_r, \theta_r, x, y, z) = (A' \circ \phi)(x_r, y_r, \theta_r, x, y, z) = A'(0, 0, \theta_r, x - x_r, y - y_r, z)$$

Dada la definición de las funciones A (??) y A' (??), se tiene que

$$A'(0, 0, \theta_r, x - x_r, y - y_r, z) = A(0, 0, \theta_r, x - x_r, y - y_r, z)$$

y por tanto

$$A(x_r, y_r, \theta_r, x, y, z) = A(0, 0, \theta_r, x - x_r, y - y_r, z) \quad (5.23)$$

De esta forma se observa que la función que define el robot permanece invariante si se efectúa su traslación respecto de los ejes (x, y) con una orientación θ_r fija, pasando de estar situado el robot en una configuración (x_r, y_r, θ_r) a estar situado en $(0, 0, \theta_r)$. Por similitud con la notación del robot en una configuración dada, se denota por

$$A_{(0,0,\theta_r)}(x - x_r, y - y_r, z) = A(0, 0, \theta_r, x - x_r, y - y_r, z)$$

Dada la estructura física del robot, se puede encontrar un sistema de coordenadas para el espacio de trabajo y para el C-espacio que permite definir un morfismo. Con éste, la función A es independiente de alguna de las coordenadas del espacio de las configuraciones. Así, con (??) y (??), se puede calcular CB de la siguiente forma

$$\begin{aligned} CB(x_r, y_r, \theta_r) &= \int A(x_r, y_r, \theta_r, x, y, z)B(x, y, z)dxdydz \\ &= \int A_{(0,0,\theta_r)}(x - x_r, y - y_r, z)B(x, y, z)dxdydz \quad (5.24) \end{aligned}$$

Esta integral se debería resolver para cada orientación θ_r del robot, es decir, para cada función $A_{(0,0,\theta_r)}$ concreta, obteniéndose un $CB(x_r, y_r, \theta_r)$ determinado. Cada uno de ellos daría lugar a un plano en un espacio de las configuraciones de tres dimensiones (x_r, y_r, θ_r) . Como ya se ha comentado, el planteamiento es similar al de la plataforma 2D que se mueve en un plano. Sin embargo, el resultado no es el mismo; para este caso se debe calcular una integral sobre tres variables, al ser tres la dimensión del espacio de trabajo.

Como se puede observar en la expresión (??), se trata del producto de convolución de dos funciones $\bar{A}_{(0,0,\theta_r)}$ y B definidas en R^3 , realizado únicamente sobre dos variables x e y . Sobre z es necesario efectuar la integral de este producto de convolución.

$$CB(x_r, y_r, \theta_r) = \int (\bar{A}_{(0,0,\theta_r)} * B)_{(x,y)}(x_r, y_r, z)dz \quad (5.25)$$

En definitiva, cada $CB(x_r, y_r, \theta_r)$ se obtiene haciendo el producto de convolución de $\bar{A}_{(0,0,\theta_r)}$ con B para cada plano $z = \text{constante}$, desde $z = 0$ hasta $z = \text{altura}_{\text{prisma}}$ y después sumando dichos productos de convolución.

Para finalizar con el método, queda aplicar el Teorema de Convolución sobre ??

$$\mathcal{F}[CB(x_r, y_r, \theta_r)] = \int \mathcal{F}[\bar{A}_{(0,0,\theta_r)}(x_r, y_r, z)]_{(x,y)} \mathcal{F}[B(x_r, y_r, z)]_{(x,y)} dz \quad (5.26)$$

Examinando esta expresión ?? se puede comprobar que los C-obstáculos se obtienen calculando su transformada de Fourier inversa. Además, hay resaltar que en ella aparece la integral sobre la coordenada z del producto de transformadas en dos dimensiones.

Capítulo 6

Aplicación del Formalismo para Robots Articulados

En el capítulo previo se ha particularizado el formalismo matemático a robots móviles que puede desplazarse libremente sobre un plano, modificando o no su orientación. En éste se va a aplicar el método matemático a robots articulados. Están formados por varios objetos rígidos que se mueven conectados por articulaciones mecánicas que restringen su movimiento relativo. Su estructura mecánica va a imponer la elección de los sistemas de referencia y las coordenadas utilizadas en el espacio de trabajo y, por tanto, va a marcar las relaciones con las coordenadas del espacio de las configuraciones.

En las secciones de este capítulo se va a aplicar el método matemático propuesto para la generación de los C-obstáculos, en primer lugar, a un manipulador planar de revolución con dos elementos y después a los manipuladores más extendidos en la industria, como son: el PUMA, el SCARA y el Stanford. Las expresiones que se obtienen van a permitir proponer algoritmos para su implementación en un ordenador.

6.1. Manipulador Planar de Revolución

Para comenzar el estudio, se va a considerar una de las estructuras mecánicas más estudiadas y más simples de robots articulados con un punto fijo: el

manipulador planar de revolución. El análisis de este manipulador ideal ayudará a comprender el de otros robots articulados, como son el PUMA y el SCARA.

Su cadena cinemática está formada por dos elementos rígidos, uno que puede girar respecto de la base y otro que se puede mover en R^2 , sujeto a las restricciones que impone el estar unido a otro elemento. En concreto, \mathbf{A} estaría formado por dos objetos rígidos \mathbf{A}_1 y \mathbf{A}_2 conectados por una articulación de revolución y otra para unir \mathbf{A}_1 a la base.

Los ejes de giro de las dos articulaciones son paralelos entre sí y perpendiculares al plano donde se asienta el robot. Con esto la elección de los sistemas de referencia para el espacio de trabajo F_W y para cada uno de los elementos del robot F_{A_1} y F_{A_2} es la siguiente (figura ??):

- F_W situado en la base del robot.
- El origen de F_{A_1} coincidente con el de F_W . Uno de los ejes de F_{A_1} se sitúa a lo largo del primer elemento, mientras que el otro es perpendicular a éste.
- El origen de F_{A_2} situado en el extremo final del elemento A_1 . Uno de los ejes situado sobre el segundo elemento y el otro perpendicular a él.

La elección de estos sistemas de referencia y de sus coordenadas asociadas va a ser crucial a la hora de poder simplificar el cálculo de $CB(q)$.

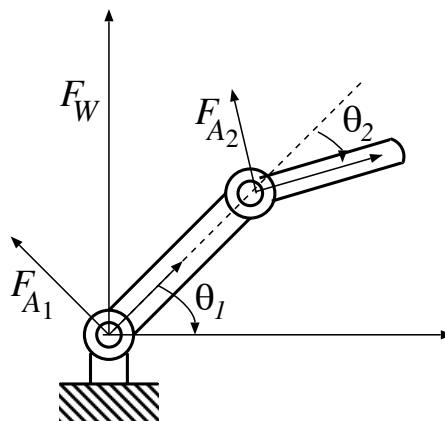


Figura 6.1: Sistemas de referencia para el manipulador planar de revolución

Una configuración de \mathbf{A} es una especificación de la la orientación de F_{A_1} y F_{A_2} con respecto a F_W . Como ya se ha comentado, su espacio de las configuraciones es de dimensión dos. La representación de C viene dada parametrizando cada configuración q por $(\theta_1, \theta_2) \in [0, 2\pi] \times [0, 2\pi]$, donde θ_1 representa la orientación de F_{A_1} con respecto a F_W , y θ_2 la orientación de F_{A_2} con respecto a F_{A_1} (figura ??).

Las coordenadas para W se van a elegir de forma tal que alguna ellas coincida con alguna de las del C espacio. Una elección posible es trabajar con coordenadas polares (r, φ) . Así las variaciones en φ están relacionadas con las variaciones en θ_1 . Esta relación va a ser clave para simplificar el cálculo de la construcción de los C-obstáculos.

Con estas consideraciones la función $B : W \rightarrow R$ estaría dada (figura ??) por

$$B(r, \varphi) = \begin{cases} 1 & \text{si } (r, \varphi) \in \mathbf{B} \\ 0 & \text{si } (r, \varphi) \notin \mathbf{B} \end{cases}$$

Dada la estructura mecánica del manipulador, la función $A : C \times W \rightarrow R$ se puede poner como suma de A_1 y de A_2 , así

$$A(\theta_1, \theta_2, r, \varphi) = A_1(\theta_1, \theta_2, r, \varphi) + A_2(\theta_1, \theta_2, r, \varphi)$$

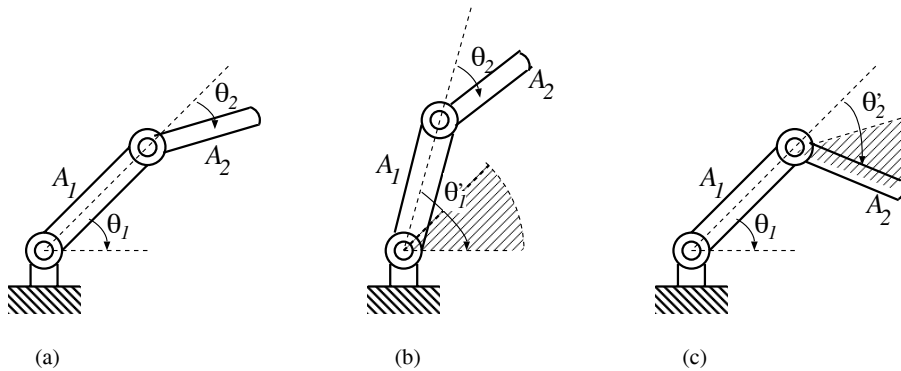


Figura 6.2: Dependencias de los movimientos de \mathbf{A}_1 y \mathbf{A}_2

En la figura ??(b) se observa cómo una variación en el ángulo de la primera articulación de θ_1 a θ_1' implica un cambio en la configuración de ambos elementos \mathbf{A}_1 y \mathbf{A}_2 . Sin embargo, si el ángulo de la segunda articulación varía de θ_2 a θ_2'

entonces sólo se producen modificaciones en la configuración del elemento \mathbf{A}_2 . Queda claro que la función A relacionada con el primer elemento A_1 no depende de θ_2 , mientras que A_2 depende tanto de θ_1 como de θ_2 . Entonces

$$A(\theta_1, \theta_2, r, \varphi) = A_1(\theta_1, r, \varphi) + A_2(\theta_1, \theta_2, r, \varphi) \quad (6.1)$$

y las dos funciones (figura ??) vendrían definidas por

$$A_1(\theta_1, r, \varphi) = \begin{cases} 1 & \text{si } (r, \varphi) \in \mathbf{A}_1(\theta_1) \\ 0 & \text{si } (r, \varphi) \notin \mathbf{A}_1(\theta_1) \end{cases} \quad (6.2)$$

$$A_2(\theta_1, \theta_2, r, \varphi) = \begin{cases} 1 & \text{si } (r, \varphi) \in \mathbf{A}_2(\theta_1, \theta_2) \\ 0 & \text{si } (r, \varphi) \notin \mathbf{A}_2(\theta_1, \theta_2) \end{cases} \quad (6.3)$$

donde $\mathbf{A}_1(\theta_1)$ es el subconjunto de puntos de W que representa al elemento \mathbf{A}_1 en la configuración θ_1 , ya que no depende de θ_2 ; y $\mathbf{A}_2(\theta_1, \theta_2)$ es el subconjunto de puntos de W que representa al elemento \mathbf{A}_2 en la configuración (θ_1, θ_2) .

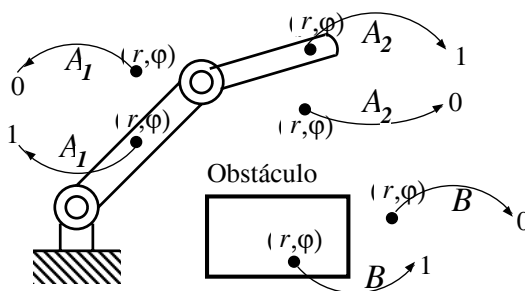


Figura 6.3: Definición de A y B para un manipulador planar de revolución

A partir de ambas definiciones $CB : C \rightarrow R$ vendría dada por

$$CB(\theta_1, \theta_2) = \int A(\theta_1, \theta_2, r, \varphi) B(r, \varphi) dr d\varphi \quad (6.4)$$

Una forma de simplificar el cálculo de esta integral es encontrar una función $A' : C' \times W' \rightarrow R$, que sea independiente de algunos de los parámetros que definen una configuración. Esto es posible gracias a la elección que se ha realizado de los sistemas de referencia y de las coordenadas. Así, si se elige $F_{W'} \equiv F_{A_1}$ (figura ??), $F_{A'_1} \equiv F_{A_1}$ y $F_{A'_2} \equiv F_{A_2}$. Respecto de estos sistema, C' contiene un conjunto de configuraciones dado por $(0, \theta'_2)$, donde $\theta'_2 = \theta_2$.

Dada la estructura cinemática de este manipulador, la función $A' : C' \times W' \rightarrow R$ se puede descomponer en dos funciones A'_1 y A'_2 , y con las mismas consideraciones que para la función A (??) se tiene que

$$A'(0, \theta'_2, r', \varphi') = A'_1(0, r', \varphi') + A'_2(0, \theta'_2, r', \varphi') \quad (6.5)$$

Las funciones A'_1 y A'_2 estarían definidas como

$$A'_1(0, r', \varphi') = \begin{cases} 1 & \text{si } (r', \varphi') \in \mathbf{A}_{\mathbf{1}(0)} \\ 0 & \text{si } (r', \varphi') \notin \mathbf{A}_{\mathbf{1}(0)} \end{cases} \quad (6.6)$$

$$A'_2(0, \theta_2, r', \varphi') = \begin{cases} 1 & \text{si } (r', \varphi') \in \mathbf{A}_{\mathbf{2}(0, \theta_2)} \\ 0 & \text{si } (r', \varphi') \notin \mathbf{A}_{\mathbf{2}(0, \theta_2)} \end{cases} \quad (6.7)$$

donde $\mathbf{A}_{\mathbf{1}(0)}$ es el subconjunto de puntos que representa al elemento $\mathbf{A}_{\mathbf{1}}$ en la configuración 0 respecto de $F_{W'}$, ya que no depende de θ_2 ; y $\mathbf{A}_{\mathbf{2}(0, \theta_2)}$ es el subconjunto de puntos de W que representa al elemento $\mathbf{A}_{\mathbf{2}}$ en la configuración $(0, \theta_2)$ respecto de $F_{W'}$. La definición de A' así como la de los subconjuntos se puede observar en la figura ??.

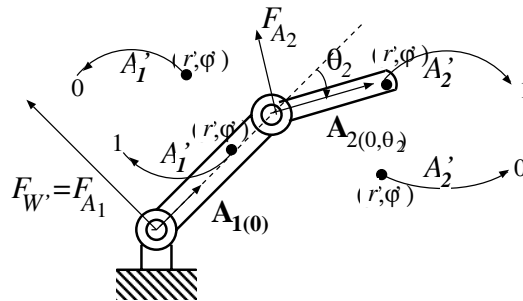


Figura 6.4: Definición de A' para un manipulador planar de revolución

Hay que definir un morfismo que efectúe esta traslación de coordenadas. La ventaja de trabajar en coordenadas polares es que los giros sobre un eje se pueden considerar que son traslaciones sobre la coordenada angular φ . El morfismo ϕ , lógicamente, vendría definido ahora por

$$\begin{aligned} \phi : C \times W &\longrightarrow C' \times W' \\ \theta_1, \theta_2, r, \varphi &\longrightarrow \theta'_1, \theta'_2, r', \varphi' \end{aligned}$$

donde

$$\begin{aligned} r' &= r & \varphi' &= \varphi - \theta_1 \\ \theta_1' &= 0 & \theta_2' &= \theta_2 \end{aligned}$$

Entonces se cumple que $A = (A' \circ \phi)$, y se obtiene que

$$A_1(\theta_1, r, \varphi) = (A_1' \circ \phi)(\theta_1, r, \varphi) = A_1'(0, r, \varphi - \theta_1)$$

$$A_2(\theta_1, \theta_2, r, \varphi) = (A_2' \circ \phi)(\theta_1, \theta_2, r, \varphi) = A_2'(0, \theta_2, r, \varphi - \theta_1)$$

Con las definiciones de funciones A (?? y ??) y A' (?? y ??), se tiene que

$$A_1'(0, r, \varphi - \theta_1) = A_1(0, r, \varphi - \theta_1)$$

$$A_2'(0, \theta_2, r, \varphi - \theta_1) = A_2(0, \theta_2, r, \varphi - \theta_1)$$

y por tanto

$$A_1(\theta_1, r, \varphi) = A_1(0, r, \varphi - \theta_1) \quad (6.8)$$

$$A_2(\theta_1, \theta_2, r, \varphi) = A_2(0, \theta_2, r, \varphi - \theta_1) \quad (6.9)$$

Como resultado se obtiene que las funciones que definen a los elementos del robot permanecen invariantes si se realiza una traslación respecto de la coordenada φ , pasando el robot de estar situado en la configuración (θ_1, θ_2) a $(0, \theta_2)$.

Por similitud con la notación del robot en una configuración dada, se denota

$$A_{1(0)}(r, \varphi - \theta_1) = A_1(0, r, \varphi - \theta_1)$$

$$A_{2(0, \theta_2)}(r, \varphi - \theta_1) = A_2(0, \theta_2, r, \varphi - \theta_1)$$

Con el sistema de coordenadas polares elegido para W y la representación (θ_1, θ_2) de C , se puede definir un morfismo que hace que el robot sea independiente del grado de libertad θ_1 . Así, con (??), (??) y (??), $CB(q)$ se puede calcular como

$$\begin{aligned} CB(\theta_1, \theta_2) &= \int A(\theta_1, \theta_2, r, \varphi) B(r, \varphi) dr d\varphi \\ &= \int (A_{1(0)}(r, \theta_1 - \varphi) + A_{2(0, \theta_2)}(r, \theta_1 - \varphi)) B(r, \varphi) dr d\varphi \end{aligned} \quad (6.10)$$

Es decir,

$$CB(\theta_1, \theta_2) = CB_1(\theta_1) + CB_2(\theta_1, \theta_2)$$

donde

$$\begin{aligned} CB_1(\theta_1) &= \int A_{1(0)}(r, \varphi - \theta_1) B(r, \varphi) dr d\varphi \\ CB_2(\theta_1, \theta_2) &= \int A_{2(0, \theta_2)}(r, \varphi - \theta_1) B(r, \varphi) dr d\varphi \end{aligned} \quad (6.11)$$

En CB_2 para cada orientación θ_2 del segundo elemento \mathbf{A}_2 del manipulador habría que efectuar la integral ?? correspondiente.

Además, se observa que se cumple la propiedad denominada *unión de elementos robóticos*, establecida en la expresión (??).

Considerando la convolución de dos funciones definidas en R^2 sobre una única variable se llega a

$$\begin{aligned} CB_1(\theta_1) &= \int (\bar{A}_{1(0)} * B)_\varphi(r, \theta_1) dr \\ CB_2(\theta_1, \theta_2) &= \int (\bar{A}_{2(0, \theta_2)} * B)_\varphi(r, \theta_1) dr \end{aligned} \quad (6.12)$$

donde $\bar{A}_{1(0)}(r, \theta_1 - \varphi) = A_{1(0)}(r, \varphi - \theta_1)$ y $\bar{A}_{2(0, \theta_2)}(r, \theta_1 - \varphi) = A_{2(0, \theta_2)}(r, \varphi - \theta_1)$. Únicamente se produce convolución en la variable θ_1 en ambas expresiones y, además, el cálculo de CB_2 debe hacerse para todos los valores posibles de θ_2 .

Si se toman transformadas de Fourier en ambos miembros de la expresión (??) se cumple que

$$\begin{aligned} \mathcal{F}[CB_1(\theta_1)] &= \int \left(\mathcal{F}[A_{1(0)}(r, \theta_1)]_\varphi \mathcal{F}[B(r, \theta_1)]_\varphi \right) dr \\ \mathcal{F}[CB_2(\theta_1, \theta_2)] &= \int \left(\mathcal{F}[A_{2(0, \theta_2)}(r, \theta_1)]_\varphi \mathcal{F}[B(r, \theta_1)]_\varphi \right) dr \end{aligned} \quad (6.13)$$

donde el subíndice φ indica que se trata de la transformada de Fourier en una única dimensión, en la coordenada angular, de dos funciones definidas en R^2 .

6.2. Robots Articulados en el Espacio

En primer lugar se va a considerar un robot articulado de dos elementos rígidos, cuyo extremo final puede moverse en R^3 (figura ??). Representa una primera aproximación a robots con un número mayor de elementos en el espacio, ya que es una mera simplificación de un PUMA. Los dos elementos \mathbf{A}_1 y \mathbf{A}_2 están conectados por una articulación de revolución, denotando su grado de libertad por θ_2 . Además, el primer elemento puede girar un ángulo θ_1 en torno a un eje perpendicular al plano donde se asienta el manipulador.

Los tres sistemas de referencia F_W , F_{A_1} y F_{A_2} se eligen (figura ??) de manera que sus orígenes situados en el punto de intersección de los dos elementos \mathbf{A}_1 y \mathbf{A}_2 . Con esta elección se ha perseguido el objetivo de situar, siempre que se pueda, los orígenes de los sistemas de referencia en los puntos de intersección de los ejes de giro consecutivos. De esta forma se evitarán desplazamientos y se obtendrán ciertas simetrías que simplificarán el cálculo de los C-obstáculos. En concreto, se han situado los sistemas de referencia de la siguiente forma (figura ??):

- El eje z de F_W se sitúa a lo largo del eje de giro del primer elemento \mathbf{A}_1 . Los ejes x e y se eligen para que formen junto con z un triedo.
- El eje z_1 del sistema de referencia F_{A_1} se hace coincidir con el eje z de F_W . Ambos sistemas de referencia coinciden cuando el primer elemento no ha girado $\theta_1 = 0$.
- El eje y_2 de F_{A_2} se coloca a lo largo del eje del segundo elemento, de forma que se mueva solidario con él. El eje x_2 , alrededor del que gira \mathbf{A}_2 , es perpendicular al plano formado por y_2 y z_1 .

Una configuración de \mathbf{A} vendría especificada por las orientaciones de F_{A_1} y F_{A_2} con respecto a F_W . Su espacio de las configuraciones sería $S^1 \times S^1$. Con el análisis efectuado en capítulos previos para un manipulador planar de revolución, cada configuración q viene parametrizada por $(\theta_1, \theta_2) \in [0, 2\pi) \times [0, 2\pi)$, donde θ_1 y θ_2 representan, respectivamente, las orientaciones de F_{A_1} y F_{A_2} con respecto a F_W . Se tiene así un espacio de las configuraciones de dimensión dos, al igual que el del manipulador planar.

Para trabajar en W se seleccionan las coordenadas esféricas $(r, \varphi_1, \varphi_2)$. Así, se consigue que el recorrido angular de alguna de las coordenadas de W coincida con el de alguna de C . Concretamente se relaciona θ_1 con φ_1 y θ_2 con φ_2 . De esta forma los giros sobre un eje se transforman en traslaciones sobre ciertas variables si se trabaja en esféricas.

Con estas consideraciones, la función $B : W \rightarrow R$ vendría definida como

$$B(r, \varphi_1, \varphi_2) = \begin{cases} 1 & \text{si } (r, \varphi_1, \varphi_2) \in \mathbf{B} \\ 0 & \text{si } (r, \varphi_1, \varphi_2) \notin \mathbf{B} \end{cases} \quad (6.14)$$

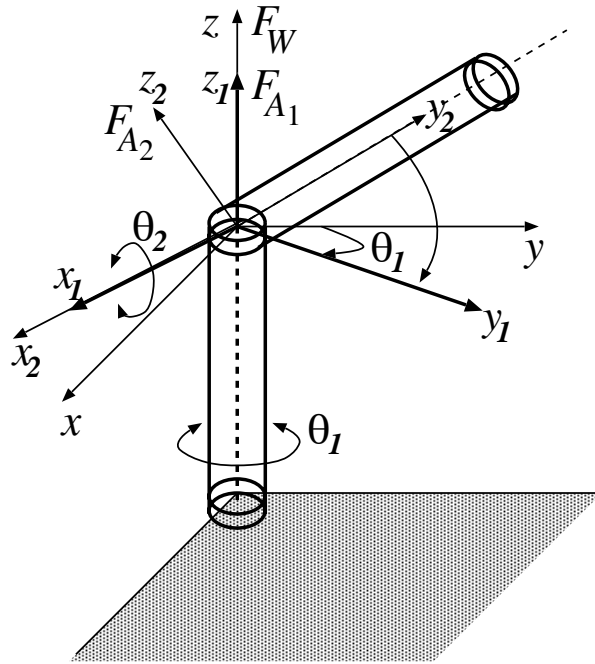


Figura 6.5: Sistemas de referencia para el manipulador reducido

donde \mathbf{B} es conjunto de puntos de W que representa a los obstáculos.

Una vez establecidos los espacios C y W , la definición de A sería

$$\begin{aligned} C \times W &\longrightarrow R \\ \theta_1, \theta_2, r, \varphi_1, \varphi_2 &\longrightarrow A(\theta_1, \theta_2, r, \varphi_1, \varphi_2) \end{aligned}$$

Por la estructura mecánica del manipulador, la función A se puede poner como

$$A(\theta_1, \theta_2, r, \varphi_1, \varphi_2) = A_1(\theta_1, \theta_2, r, \varphi_1, \varphi_2) + A_2(\theta_1, \theta_2, r, \varphi_1, \varphi_2) \quad (6.15)$$

Es claro que el primer elemento \mathbf{A}_1 del manipulador no depende de θ_2 , mientras que el segundo elemento depende de los dos grados de libertad. Entonces

$$A(\theta_1, \theta_2, r, \varphi_1, \varphi_2) = A_1(\theta_1, r, \varphi_1, \varphi_2) + A_2(\theta_1, \theta_2, r, \varphi_1, \varphi_2) \quad (6.16)$$

Las funciones A_1 y A_2 vendrían definidas por

$$A_1(\theta_1, r, \varphi_1, \varphi_2) = \begin{cases} 1 & \text{si } (r, \varphi_1, \varphi_2) \in \mathbf{A}_1(\theta_1) \\ 0 & \text{si } (r, \varphi_1, \varphi_2) \notin \mathbf{A}_1(\theta_1) \end{cases} \quad (6.17)$$

$$A_2(\theta_1, \theta_2, r, \varphi_1, \varphi_2) = \begin{cases} 1 & \text{si } (r, \varphi_1, \varphi_2) \in \mathbf{A}_2(\theta_1, \theta_2) \\ 0 & \text{si } (r, \varphi_1, \varphi_2) \notin \mathbf{A}_2(\theta_1, \theta_2) \end{cases} \quad (6.18)$$

donde

- $\mathbf{A}_{1(\theta_1)}$ es el subconjunto de puntos de W que representa al primer elemento de la cadena cinemática en la configuración θ_1 , pues no depende de θ_2 .
- $\mathbf{A}_{2(\theta_1, \theta_2)}$ es el subconjunto de puntos de W que representa al elemento \mathbf{A}_2 en la configuración (θ_1, θ_2) .

Como consecuencia, la función $CB : C \rightarrow R$ vendría definida como

$$CB(\theta_1, \theta_2) = \int A(\theta_1, \theta_2, r, \varphi_1, \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \quad (6.19)$$

Sustituyendo en (??) la expresión (??) se obtiene

$$CB(\theta_1, \theta_2) = CB_1(\theta_1) + CB_2(\theta_1, \theta_2)$$

donde

$$\begin{aligned} CB_1(\theta_1) &= \int A_1(\theta_1, r, \varphi, \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \\ CB_2(\theta_1, \theta_2) &= \int A_2(\theta_1, \theta_2, r, \varphi_1, \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \end{aligned} \quad (6.20)$$

Estas son las expresiones que hay que calcular para tener representados los obstáculos en el espacio de las configuraciones. A continuación se pretende reducir, en lo posible, el cálculo que estas integrales conllevan. Se van a considerar de forma independiente cada una de ellas y, por tanto, hay que encontrar unas funciones $A'_1 : C' \times W' \rightarrow R$ y $A'_2 : C'' \times W'' \rightarrow R$, que sean independientes de alguno de los parámetros que definen una configuración. Esto es posible si se eligen de forma adecuada los sistemas de referencia y de coordenadas. Así se ha realizado la siguiente elección (figura ??):

- $F_{W'} \equiv F_{A_1}$
- $F_{W''} \equiv F_{A_2}$

Una configuración vendría especificada por la orientación de F_{A_1} respecto de $F_{W'}$ y de F_{A_2} respecto de $F_{W''}$. Respecto de estos sistemas, C' contiene una única configuración dada por $q = (0, 0)$. Con las consideraciones realizadas A'_1 y A'_2 estarían definidas (figura ??) como

$$A'_1(0, r', \varphi'_1, \varphi'_2) = \begin{cases} 1 & \text{si } (r', \varphi'_1, \varphi'_2) \in \mathbf{A}_{1(0)} \\ 0 & \text{si } (r', \varphi'_1, \varphi'_2) \notin \mathbf{A}_{1(0)} \end{cases} \quad (6.21)$$

$$A'_2(0,0,r',\varphi'_1,\varphi'_2) = \begin{cases} 1 & \text{si } (r',\varphi'_1,\varphi'_2) \in \mathbf{A}_2(0,0) \\ 0 & \text{si } (r',\varphi'_1,\varphi'_2) \notin \mathbf{A}_2(0,0) \end{cases} \quad (6.22)$$

siendo

- $\mathbf{A}_1(0)$ es el subconjunto de puntos que representa al elemento \mathbf{A}_1 en la configuración 0 respecto de $F_{W'}$, ya que no depende de θ_2 .
- $\mathbf{A}_2(0,0)$ es el subconjunto de puntos de W que representa al elemento \mathbf{A}_2 en la configuración $(0,0)$ respecto de $F_{W''}$.

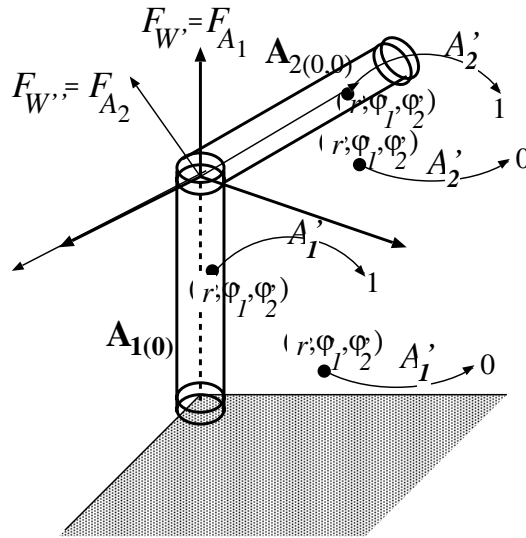


Figura 6.6: Definición de A' para un manipulador reducido de revolución

Al haber elegido las esféricas como coordenadas de trabajo, el cambio de los sistemas de referencia iniciales a los nuevos es una traslación. Realmente la ventaja que supone trabajar con esféricas es que los giros θ_1 y θ_2 sobre los dos ejes son traslaciones sobre las coordenadas angulares φ_1 y φ_2 . Dado que se está considerando de forma separada los dos elementos del manipulador, hay que definir dos morfismos ϕ_1 y ϕ_2 que efectúen estos cambios de sistemas de referencia. El morfismo ϕ_1 , lógicamente, vendría definido ahora por

$$\begin{aligned} C \times W &\longrightarrow C' \times W' \\ \theta_1, \theta_2, r, \varphi_1, \varphi_2 &\longrightarrow \theta'_1, \theta'_2, r', \varphi'_1, \varphi'_2 \end{aligned}$$

donde

$$\begin{aligned} r' &= r \\ \varphi'_1 &= \varphi_1 - \theta_1 & \varphi'_2 &= \varphi_2 \\ \theta'_1 &= 0 & \theta'_2 &= \theta_2 \end{aligned}$$

mientras que el morfismo ϕ_2 se definiría como

$$\begin{aligned} C \times W &\longrightarrow C'' \times W'' \\ \theta_1, \theta_2, r, \varphi_1, \varphi_2 &\longrightarrow \theta''_1, \theta''_2, r'', \varphi''_1, \varphi''_2 \end{aligned}$$

donde

$$\begin{aligned} r'' &= r \\ \varphi''_1 &= \varphi_1 - \theta_1 & \varphi''_2 &= \varphi_2 - \theta_2 \\ \theta''_1 &= 0 & \theta''_2 &= 0 \end{aligned}$$

Entonces se cumple que $A_1 = (A'_1 \circ \phi_1)$ y $A_2 = (A'_2 \circ \phi_2)$, y se obtendría que

$$A_1(\theta_1, r, \varphi_1, \varphi_2) = (A'_1 \circ \phi_1)(\theta_1, r, \varphi_1, \varphi_2) = A'_1(0, r, \varphi_1 - \theta_1, \varphi_2)$$

$$A_2(\theta_1, \theta_2, r, \varphi_1, \varphi_2) = (A'_2 \circ \phi)(\theta_1, \theta_2, r, \varphi_1, \varphi_2) = A'_2(0, 0, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2)$$

Con las definiciones de las funciones A (?? y ??) y A' (?? y ??), se tiene que

$$A'_1(0, r, \varphi_1 - \theta_1, \varphi_2) = A_1(0, r, \varphi_1 - \theta_1, \varphi_2)$$

$$A'_2(0, 0, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) = A_2(0, 0, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2)$$

y por tanto

$$A_1(\theta_1, r, \varphi_1, \varphi_2) = A_1(0, r, \varphi_1 - \theta_1, \varphi_2) \quad (6.23)$$

$$A_2(\theta_1, \theta_2, r, \varphi_1, \varphi_2) = A_2(0, 0, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \quad (6.24)$$

Como resultado se obtiene que las funciones que definen a los elementos del robot permanecen invariantes si se realiza una traslación respecto de las coordenadas φ_1 y φ_2 , pasando el robot de estar situado en la configuración $q = (\theta_1, \theta_2)$ a encontrarse en $q = (0, 0)$. Por similitud con la notación del robot en una configuración dada, se denota por

$$A_{1(0)}(r, \varphi_1 - \theta_1, \varphi_2) = A_1(0, r, \varphi_1 - \theta_1, \varphi_2)$$

$$A_{2(0,0)}(r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) = A_2(0, 0, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2)$$

Sustituyendo (??) y (??) en (??), $CB(q)$ vendría dado por

$$CB(\theta_1, \theta_2) = CB_1(\theta_1) + CB_2(\theta_1, \theta_2)$$

donde

$$\begin{aligned} CB_1(\theta_1) &= \int A_{1(0)}(r, \varphi_1 - \theta_1, \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \\ CB_2(\theta_1, \theta_2) &= \int A_{2(0,0)}(r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \end{aligned} \quad (6.25)$$

Se supone que no existe colisión entre el elemento \mathbf{A}_1 del manipulador y algún obstáculo en W , ya que si no el robot no podría moverse. Entonces, por definición, el primer término de la expresión previa (??) es nulo, y como consecuencia

$$CB(\theta_1, \theta_2) = \int A_{2(0,0)}(r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \quad (6.26)$$

En este planteamiento existen dos elecciones clave: las coordenadas esféricas como coordenadas de trabajo para W y (θ_1, θ_2) como la representación de C . Esto ha permitido definir un morfismo que hace que el robot sea independiente de los grados de libertad θ_1 y θ_2 . Así, se consigue el objetivo perseguido con este planteamiento, que era simplificar el cálculo de $CB(q)$.

Considerando la convolución de dos funciones definidas en R^3 sobre dos de variables se llega a θ_1 y θ_2 .

$$CB_2(\theta_1, \theta_2) = \int (\bar{A}_{2(0,0)} * B)_{(\varphi_1, \varphi_2)}(r, \theta_1, \theta_2) dr \quad (6.27)$$

donde

$$\bar{A}_{2(0,0)}(r, \theta_1 - \varphi_1, \theta_2 - \varphi_2) = A_{2(0,0)}(r, \varphi_1 - \theta_1, \varphi_2 - \theta_2)$$

Por tanto, si se toman transformadas de Fourier en dos dimensiones sobre ambos miembros de la expresión (??) se cumple que

$$\mathcal{F}[CB(\theta_1, \theta_2)] = \int \mathcal{F}[\bar{A}_{2(0,0)}(r, \theta_1, \theta_2)]_{(\varphi_1, \varphi_2)} \mathcal{F}[B(r, \theta_1, \theta_2)]_{(\varphi_1, \varphi_2)} dr \quad (6.28)$$

donde el subíndice (φ_1, φ_2) indica que se trata de transformadas de Fourier en dos dimensiones de funciones definidas en R^3 .

Este ejemplo se ha estudiado como un caso hipotético que ayudará abordar el estudio del robot PUMA y del manipulador Stanford de una forma más fácil. La extensión de este ejemplo a los otros dos supone añadir un elemento más a su cadena cinemática, con una articulación de revolución para el PUMA y una prismática para el Stanford. Además, ha permitido trabajar con espacios de trabajo de R^3 y con espacios de las configuraciones de dimensión reducida.

6.3. Robot PUMA

Un tipo de robot articulado ampliamente utilizado en la industria es el PUMA. Está formado por tres elementos rígidos \mathbf{A}_1 , \mathbf{A}_2 y \mathbf{A}_3 conectados por dos articulaciones de revolución, denominadas hombro y codo, siendo sus grados de libertad θ_2 y θ_3 , respectivamente (figura ??). Además, el primer elemento puede girar alrededor de un eje perpendicular al plano donde se asienta el manipulador (cintura), denominándose su grado de libertad θ_1 . El eje de giro de la articulación codo es paralelo al del hombro, y ambos son perpendiculares al eje de la cintura.

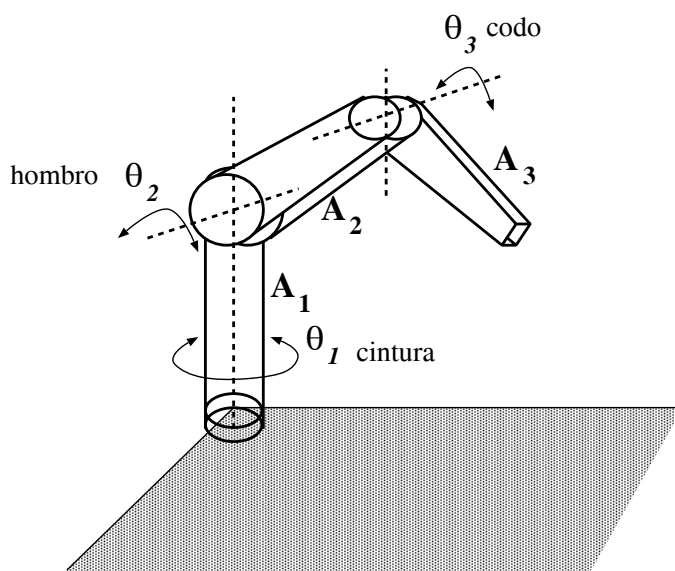


Figura 6.7: Elementos y articulaciones del manipulador PUMA

Teniendo presente esta disposición, se elegirán los orígenes de los sistemas de referencia, con el criterio que ya se ha comentado, para simplificar el cálculo de $CB(q)$. En concreto, se ha seguido el siguiente procedimiento:

- Los tres sistemas de referencia F_W , F_{A_1} y F_{A_2} , asociados respectivamente con el espacio de trabajo W y con los dos primeros elementos \mathbf{A}_1 y \mathbf{A}_2 , se eligen de forma análoga al robot reducido.
- Se hacen coincidir los tres orígenes y se sitúan sobre la articulación

del hombro. En esta articulación intersectan los ejes de giro de los dos primeros elementos.

- El eje z de F_W se sitúa a lo largo del eje del primer elemento \mathbf{A}_1 . Los ejes x e y se eligen para que formen junto con z un triedo.
 - El eje z_1 del sistema de referencia F_{A_1} se hace coincidir con el eje z de F_W . Ambos sistemas de referencia coinciden cuando el primer elemento no ha girado $\theta_1 = 0$.
 - El eje y_2 de F_{A_2} se coloca a lo largo del eje del segundo elemento, de forma que se mueva solidario con él. El eje x_2 , alrededor del que gira \mathbf{A}_2 , es perpendicular al plano formado por y_2 y z_1 .
- Respecto al sistema de referencia F_{A_3} asociado al tercer elemento, su origen se sitúa sobre la articulación codo. El eje y_3 se coloca a lo largo del eje de \mathbf{A}_3 . El eje x_3 , alrededor del que gira el tercer elemento, es perpendicular al plano formado por los vectores y_2 e y_3 .

Una configuración de A vendría especificada por la orientación de F_{A_1} y F_{A_2} con respecto a F_W y la orientación de F_{A_3} con respecto a F_{A_2} . Su espacio de las configuraciones sería $S^1 \times S^1 \times S^1$. Como ya se ha comentado, cada configuración q vendría parametrizada por $(\theta_1, \theta_2, \theta_3) \in [0, 2\pi) \times [0, 2\pi) \times [0, 2\pi)$, donde θ_1 y θ_2 representan, respectivamente, las orientaciones de F_{A_1} y F_{A_2} con respecto a F_W , y θ_3 representa la orientación de F_{A_3} con respecto a F_{A_2} . Las coordenadas del espacio de las configuraciones son los tres grados de libertad para las articulaciones: cintura, hombro y codo.

De la misma forma que en el robot de dos elementos en R^3 , se seleccionan las coordenadas esféricas $(r, \varphi_1, \varphi_2)$ como coordenadas de trabajo para W . Así, el recorrido angular de θ_1 y θ_2 se corresponde con el de φ_1 y φ_2 . Por tanto, es válida la definición ?? de la función B que se propuso para el caso simplificado.

Una vez establecidos los espacios C y W , la definición de A sería

$$\begin{aligned} A : C \times W &\longrightarrow R \\ \theta_1, \theta_2, \theta_3, r, \varphi_1, \varphi_2 &\longrightarrow A(\theta_1, \theta_2, \theta_3, r, \varphi_1, \varphi_2) \end{aligned}$$

Teniendo en cuenta que el manipulador PUMA se construye añadiendo un elemento \mathbf{A}_3 al manipulador reducido, en la expresión (??) habría que introducir

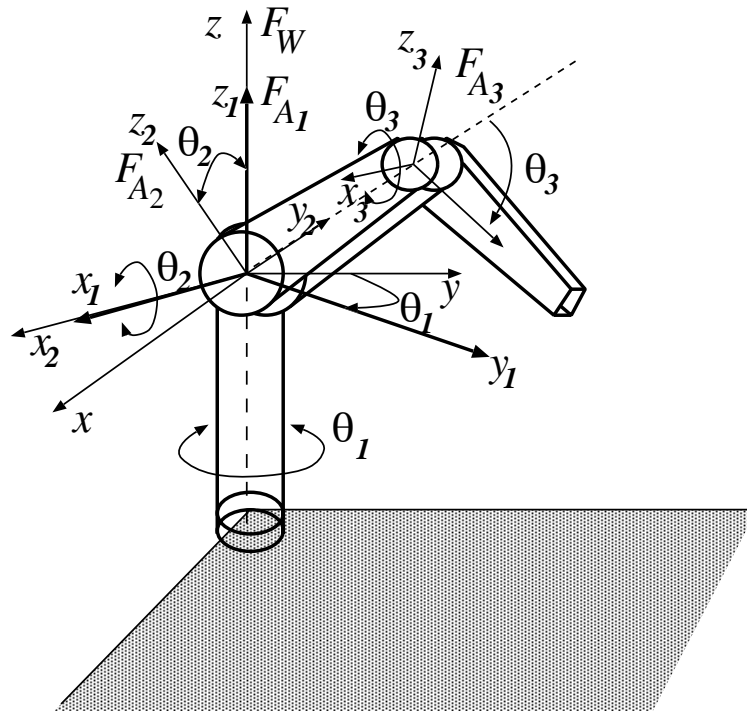


Figura 6.8: Sistemas de referencia para el manipulador PUMA

un nuevo término $A_3(\theta_1, \theta_2, \theta_3, r, \varphi_1, \varphi_2)$ Como se justificó para el robot simplificado, el primer elemento \mathbf{A}_1 no presenta dependencia con θ_2 y θ_3 . Además, el segundo elemento \mathbf{A}_2 no depende de θ_3 , mientras que el tercer elemento depende de los tres grados de libertad. Por tanto

$$A(\theta_1, \theta_2, \theta_3, r, \varphi_1, \varphi_2) = A_1(\theta_1, r, \varphi_1, \varphi_2) + A_2(\theta_1, \theta_2, r, \varphi_1, \varphi_2) + A_3(\theta_1, \theta_2, \theta_3, r, \varphi_1, \varphi_2) \quad (6.29)$$

Las dos funciones A_1 y A_2 estarían definidas de la misma forma que para el manipulador reducido (expresiones ?? y ??), mientras que A_3 estaría definida de la siguiente forma

$$A_3(\theta_1, \theta_2, \theta_3, r, \varphi_1, \varphi_2) = \begin{cases} 1 & \text{si } (r, \varphi_1, \varphi_2) \in \mathbf{A}_3(\theta_1, \theta_2, \theta_3) \\ 0 & \text{si } (r, \varphi_1, \varphi_2) \notin \mathbf{A}_3(\theta_1, \theta_2, \theta_3) \end{cases} \quad (6.30)$$

donde $\mathbf{A}_3(\theta_1, \theta_2, \theta_3)$ es el subconjunto de puntos de W que representa al elemento \mathbf{A}_3 en la configuración $(\theta_1, \theta_2, \theta_3)$.

Como consecuencia, para proyectar los obstáculos en el espacio de las configuraciones es necesario resolver la integral siguiente

$$CB(\theta_1, \theta_2, \theta_3) = \int A(\theta_1, \theta_2, \theta_3, r, \varphi_1, \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \quad (6.31)$$

Como ya se ha explicado, hay que encontrar unas funciones A'_1 , A'_2 y A'_3 que no tengan una dependencia explícita con alguno de los parámetros que definen una configuración, aprovechando la relación que existe entre el recorrido angular de las coordenadas del espacio de las configuraciones y las elegidas para el espacio de trabajo. Esto es posible si se hace una elección de los sistemas de referencia para la definición de A'_1 y otra para A'_2 y A'_3 (figura ??). La elección para la primera es

- $F_{W'} \equiv F_{A_1}$ y $F_{A'_1} \equiv F_{A_1}$
- Sin modificación los sistemas F_{A_2} y F_{A_3}

y para las segundas

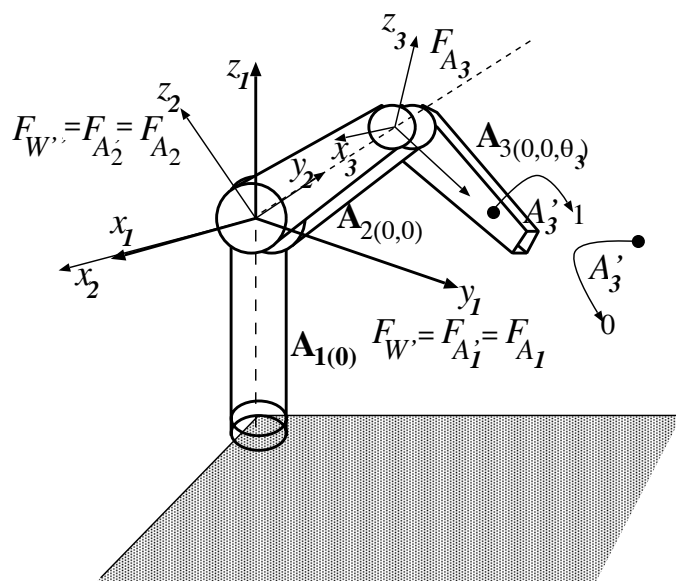
- $F_{W''} \equiv F_{A_2}$ y $F_{A'_2} \equiv F_{A_2}$
- $F_{A'_3} \equiv F_{A_3}$

Se han definido dos sistemas de referencia para el espacio de trabajo, $F_{W'}$ y $F_{W''}$, asociados respectivamente con la primera y la segunda articulación. Respecto de estos nuevos sistemas se consigue que el primer y el segundo elemento permanezcan fijos. Así, una configuración vendría dada especificando la orientación de $F_{A'_1}$ respecto a $F_{W'}$ y de $F_{A'_2}$ y de $F_{A'_3}$ respecto a $F_{W''}$. Entonces, C' contiene un conjunto de configuraciones dado por $(0, 0, \theta'_3)$, donde $\theta'_3 = \theta_3$. Las funciones A'_1 y A'_2 estarían definidas como para el manipulador reducido (?? y ??), y la función A'_3 vendría definida por

$$A'_3(0, 0, \theta'_3, r', \varphi'_1, \varphi'_2) = \begin{cases} 1 & \text{si } (r', \varphi'_1, \varphi'_2) \in \mathbf{A}_2(0, 0, \theta'_3) \\ 0 & \text{si } (r', \varphi'_1, \varphi'_2) \notin \mathbf{A}_2(0, 0, \theta'_3) \end{cases} \quad (6.32)$$

siendo $\mathbf{A}_3(0, 0, \theta_3)$ es el subconjunto de puntos de W que representa al elemento \mathbf{A}_3 en la configuración $(0, 0, \theta_3)$ respecto de $F_{W''}$ (figura ??).

El paso de los sistemas de referencia iniciales a los nuevos es una traslación de coordenadas, debido a que se ha elegido trabajar con coordenadas esféricas.

Figura 6.9: Definición de A' para un manipulador PUMA

De esta forma, se tiene que los giros θ_1 y θ_2 sobre los ejes z_1 y x_2 se convierten en traslaciones sobre las coordenadas angulares φ_1 y φ_2 . Sin embargo, no existe una relación entre θ_3 y alguna de las variables del espacio de trabajo. Con esto, es posible definir dos morfismos ϕ_1 y ϕ_2 , donde se efectúen estos cambios de sistemas de referencia. La definición de ϕ_1 estaría dada por

$$\begin{aligned} \phi_1 : C \times W &\longrightarrow C' \times W' \\ \theta_1, \theta_2, \theta_3, r, \varphi_1, \varphi_2 &\longrightarrow \theta'_1, \theta'_2, \theta'_3, r', \varphi'_1, \varphi'_2 \end{aligned}$$

donde

$$\begin{aligned} r' &= r & \varphi'_1 &= \varphi_1 - \theta_1 \\ \varphi'_2 &= \varphi_2 & \theta'_1 &= 0 \\ \theta'_2 &= \theta_2 & \theta'_3 &= \theta_3 \end{aligned}$$

Y ϕ_2 se definiría como

$$\begin{aligned} \phi_2 : C \times W &\longrightarrow C'' \times W'' \\ \theta_1, \theta_2, \theta_3, r, \varphi_1, \varphi_2 &\longrightarrow \theta''_1, \theta''_2, \theta''_3, r'', \varphi''_1, \varphi''_2 \end{aligned}$$

donde

$$\begin{aligned} r'' &= r & \varphi''_1 &= \varphi_1 - \theta_1 \\ \varphi''_2 &= \varphi_2 - \theta_2 & \theta''_1 &= 0 \\ \theta''_2 &= 0 & \theta''_3 &= \theta_3 \end{aligned}$$

Entonces se cumple que $A_1 = (A'_1 \circ \phi_1)$, $A_2 = (A'_2 \circ \phi_2)$ y $A_3 = (A'_3 \circ \phi_2)$, y se obtendría que

$$\begin{aligned} A_1(\theta_1, r, \varphi_1, \varphi_2) &= A'_1(0, r, \varphi_1 - \theta_1, \varphi_2) \\ A_2(\theta_1, \theta_2, r, \varphi_1, \varphi_2) &= A'_2(0, 0, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \\ A_3(\theta_1, \theta_2, \theta_3, r, \varphi_1, \varphi_2) &= A'_3(0, 0, \theta_3, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \end{aligned}$$

Con las definiciones de las funciones A (??, ?? y ??) y A' (??, ?? y ??), se tiene que

$$\begin{aligned} A'_1(0, r, \varphi_1 - \theta_1, \varphi_2) &= A_1(0, r, \varphi_1 - \theta_1, \varphi_2) \\ A'_2(0, 0, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) &= A_2(0, 0, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \\ A'_3(0, 0, \theta_3, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) &= A_3(0, 0, \theta_3, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \end{aligned}$$

y por tanto

$$A_1(\theta_1, r, \varphi_1, \varphi_2) = A_1(0, r, \varphi_1 - \theta_1, \varphi_2) \quad (6.33)$$

$$A_2(\theta_1, \theta_2, r, \varphi_1, \varphi_2) = A_2(0, 0, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \quad (6.34)$$

$$A_3(\theta_1, \theta_2, \theta_3, r, \varphi_1, \varphi_2) = A_3(0, 0, \theta_3, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \quad (6.35)$$

Como resultado se obtiene que las funciones que definen a los elementos del robot permanecen invariantes si se realiza una traslación respecto de las coordenadas φ_1 y φ_2 , pasando el robot de estar situado en la configuración $q = (\theta_1, \theta_2, \theta_3)$ a encontrarse en $q = (0, 0, \theta_3)$.

Si se toma la misma notación que para el manipulador reducido junto con

$$A_{3(0,0,\theta_3)}(r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) = A_3(0, 0, \theta_3, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2)$$

Sustituyendo (??), (??) y (??) en ??, $CB(q)$ vendría dado por

$$CB(\theta_1, \theta_2, \theta_3) = CB_1(\theta_1) + CB_2(\theta_1, \theta_2) + CB_3(\theta_1, \theta_2, \theta_3)$$

donde

$$\begin{aligned} CB_1(\theta_1) &= \int A_{1(0)}(r, \varphi_1 - \theta_1, \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \quad (6.36) \\ CB_2(\theta_1, \theta_2) &= \int A_{2(0,0)}(r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \\ CB_3(\theta_1, \theta_2, \theta_3) &= \int A_{3(0,0,\theta_3)}(r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \end{aligned}$$

Como se puede observar se cumple la propiedad denominada *unión de elementos robóticos*, establecida en la expresión ??.

Suponiendo que no existe colisión entre el elemento \mathbf{A}_1 del manipulador y algún obstáculo en B , entonces $CB_1 = 0$, y como consecuencia

$$CB(\theta_1, \theta_2, \theta_3) = CB_2(\theta_1, \theta_2) + CB_3(\theta_1, \theta_2, \theta_3)$$

En CB_3 para cada orientación θ_3 del tercer elemento A_3 del manipulador habría que efectuar la integral (??) correspondiente.

Considerando la convolución de dos funciones definidas en R^3 sobre dos de las variables se llega a

$$\begin{aligned} CB_2(\theta_1, \theta_2) &= \int (\bar{A}_{2(0,0)} * B)_{(\varphi_1, \varphi_2)}(r, \theta_1, \theta_2) dr \\ CB_3(\theta_1, \theta_2, \theta_3) &= \int (\bar{A}_{3(0,0,\theta_3)} * B)_{(\varphi_1, \varphi_2)}(r, \theta_1, \theta_2) dr \end{aligned} \quad (6.37)$$

donde

$$\begin{aligned} \bar{A}_{2(0,0)}(r, \theta_1 - \varphi_1, \theta_2 - \varphi_2) &= A_{2(0,0)}(r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \\ \bar{A}_{3(0,0,\theta_3)}(r, \theta_1 - \varphi_1, \theta_2 - \varphi_2) &= A_{3(0,0,\theta_3)}(r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \end{aligned}$$

En este caso se produce convolución sobre dos de las variables θ_1 y θ_2 . Por tanto, si se toman transformadas de Fourier en dos dimensiones sobre ambos miembros de la expresión ?? se cumple que

$$\mathcal{F}[CB_2] = \int \mathcal{F}[A_{2(0,0)}(r, \theta_1, \theta_2)]_{(\varphi_1, \varphi_2)} \mathcal{F}[B(r, \theta_1, \theta_2)]_{(\varphi_1, \varphi_2)} dr \quad (6.38)$$

$$\mathcal{F}[CB_3] = \int \mathcal{F}[A_{3(0,0,\theta_3)}(r, \theta_1, \theta_2)]_{(\varphi_1, \varphi_2)} \mathcal{F}[B(r, \theta_1, \theta_2)]_{(\varphi_1, \varphi_2)} dr \quad (6.39)$$

donde el subíndice (φ_1, φ_2) indica que se trata de transformadas de Fourier en dos dimensiones de dos funciones definidas en R^3 .

Como se habrá podido observar, el análisis efectuado para el cálculo de $CB(q)$ de un PUMA con tres articulaciones está estrechamente relacionado con el manipulador planar y con el robot articulado de dos elementos rígidos en R^3 . Esto es evidente ya que la cadena cinemática del PUMA se puede ver como la composición de la de ambos.

6.4. Robot SCARA

El robot denominado SCARA está diseñado para realizar, específicamente, tareas de montaje. Su cadena cinemática consta de tres elementos rígidos \mathbf{A}_1 , \mathbf{A}_2 y \mathbf{A}_3 conectados por dos articulaciones: una de revolución y otra prismática, denominándose a sus grados de libertad θ_2 y d (figura ??). Además, el primer elemento puede girar alrededor de un eje perpendicular al plano donde se asienta el manipulador, denotándose su grado de libertad por θ_1 .

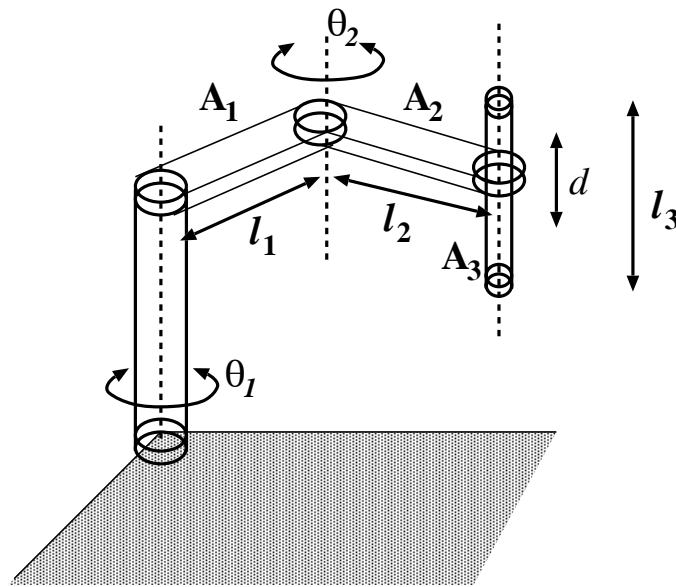


Figura 6.10: Elementos y articulaciones del manipulador SCARA

Como se puede observar en la figura ??, los tres ejes de las articulaciones son paralelos entre sí y perpendiculares al plano donde se asienta el robot. Por tanto, la elección, que se ha realizado, de los sistemas de referencia (??) es la siguiente:

- Los orígenes de F_W y F_{A_1} se colocan ambos sobre el extremo inicial del elemento \mathbf{A}_1 . Además, se hacen coincidir los ejes z de ambos sistemas de referencia con el eje de giro del primer elemento \mathbf{A}_1 . El eje y_1 se dispone a lo largo de dicho elemento.
- El origen de F_{A_2} se sitúa sobre la articulación que conecta los elementos

\mathbf{A}_1 y \mathbf{A}_2 . El eje z_2 coincide con el eje de giro de esta articulación y el eje y_2 se coloca sobre el segundo elemento.

- El sistema F_{A_3} asociado al tercer elemento se desplaza según lo hace \mathbf{A}_3 y su origen se ha colocado sobre el extremo superior de ese elemento. El eje z_3 se hace coincidir con el eje sobre el que se desplaza \mathbf{A}_3 . El eje y_3 se elige en la misma dirección que y_2 .

Una configuración de A vendría especificada por la orientación de F_{A_1} y F_{A_2} con respecto a F_W y por el desplazamiento de F_{A_3} con respecto a F_W . Su espacio de las configuraciones sería $S_1 \times S_1 \times R$ y un punto de este espacio q vendría parametrizado por $(\theta_1, \theta_2, d) \in [0, 2\pi) \times [0, 2\pi) \times [0, l_3)$, donde se denota con θ_1 y θ_2 , respectivamente, a las orientaciones de F_{A_1} y F_{A_2} con respecto a F_W , con d al desplazamiento de F_{A_3} con respecto a F_W y con l_3 a la longitud del tercer elemento. En definitiva, los parámetros que definen una configuración son los grados de libertad para las tres articulaciones.

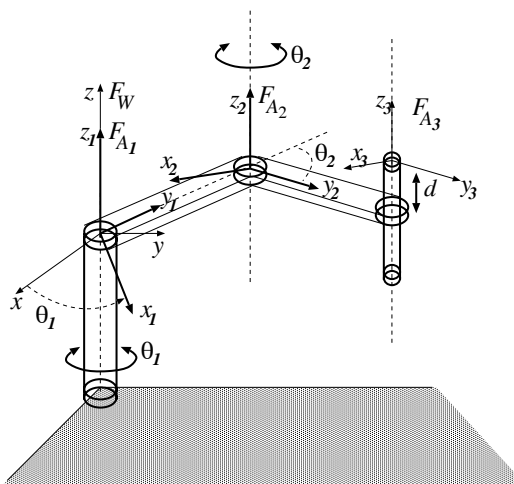


Figura 6.11: Definición de sistemas de referencia para el SCARA

En el espacio de trabajo W se eligen las coordenadas cilíndricas (r, φ, z) . De esta forma, el recorrido angular de θ_1 está relacionado con φ y el desplazamiento d está asociado con z .

Por tanto, la definición de la función $B : W \rightarrow R$ vendría definida como

$$B(r, \varphi, z) = \begin{cases} 1 & \text{si } (r, \varphi, z) \in \mathbf{B} \\ 0 & \text{si } (r, \varphi, z) \notin \mathbf{B} \end{cases} \quad (6.40)$$

donde \mathbf{B} es conjunto de puntos de W que representa a los obstáculos.

La definición de $A : C \times W \rightarrow R$ sería

$$A(\theta_1, \theta_2, d, r, \varphi, z) = A_1(\theta_1, \theta_2, d, r, \varphi, z) + A_2(\theta_1, \theta_2, d, r, \varphi, z) + A_3(\theta_1, \theta_2, d, r, \varphi, z)$$

A continuación se analizan las dependencias de cada uno de los términos con los tres grados de libertad:

- \mathbf{A}_1 no presenta dependencia con θ_2 y d .
- \mathbf{A}_2 no depende del desplazamiento d del tercer elemento.
- \mathbf{A}_3 depende de los dos giros θ_1 y θ_2 y del desplazamiento d_3 .

Por tanto

$$A(\theta_1, \theta_2, d, r, \varphi, z) = A_1(\theta_1, r, \varphi, z) + A_2(\theta_1, \theta_2, r, \varphi, z) + A_3(\theta_1, \theta_2, d_3, r, \varphi, z) \quad (6.41)$$

Como se han hecho coincidir los orígenes de F_W y F_{A_1} , los elementos \mathbf{A}_1 y \mathbf{A}_2 se encuentran en un plano $z = 0$, y entonces

$$A(\theta_1, \theta_2, d, r, \varphi, z) = A_1(\theta_1, r, \varphi)\delta(z) + A_2(\theta_1, \theta_2, r, \varphi)\delta(z) + A_3(\theta_1, \theta_2, d, r, \varphi, z) \quad (6.42)$$

donde $\delta(z)$ es la función delta de Dirac.

Las funciones en que se ha desglosado la función A estarían definidas por

$$A_1(\theta_1, r, \varphi) = \begin{cases} 1 & \text{si } (r, \varphi) \in \mathbf{A}_1(\theta_1) \\ 0 & \text{si } (r, \varphi) \notin \mathbf{A}_1(\theta_1) \end{cases} \quad (6.43)$$

$$A_2(\theta_1, \theta_2, r, \varphi) = \begin{cases} 1 & \text{si } (r, \varphi) \in \mathbf{A}_2(\theta_1, \theta_2) \\ 0 & \text{si } (r, \varphi) \notin \mathbf{A}_2(\theta_1, \theta_2) \end{cases} \quad (6.44)$$

$$A_3(\theta_1, \theta_2, d, r, \varphi, z) = \begin{cases} 1 & \text{si } (r, \varphi, z) \in \mathbf{A}_3(\theta_1, \theta_2, d) \\ 0 & \text{si } (r, \varphi, z) \notin \mathbf{A}_3(\theta_1, \theta_2, d) \end{cases} \quad (6.45)$$

donde

- $\mathbf{A}_{1(\theta_1)}$ es el subconjunto de puntos de W que representa al primer elemento de la cadena cinemática en la configuración θ_1 , independientemente de θ_2 y d .
- $\mathbf{A}_{2(\theta_1, \theta_2)}$ es el subconjunto de puntos de W que representa al elemento \mathbf{A}_2 en la configuración (θ_1, θ_2) , pues no depende de d .
- $\mathbf{A}_{3(\theta_1, \theta_2, d)}$ es el subconjunto de W que representa al tercer elemento en la configuración (θ_1, θ_2, d) .

Con estas consideraciones la función CB de C en R estaría definida como

$$CB(\theta_1, \theta_2, d) = \int A(\theta_1, \theta_2, d, r, \varphi, z)B(r, \varphi, z)drd\varphi dz \quad (6.46)$$

Sustituyendo la expresión (??) en (??) entonces

$$CB(\theta_1, \theta_2, d) = \int (A_1(\theta_1, r, \varphi)\delta(z) + A_2(\theta_1, \theta_2, r, \varphi)\delta(z) + A_3(\theta_1, \theta_2, d, r, \varphi, z))B(r, \varphi, z)drd\varphi dz \quad (6.47)$$

Esta expresión se puede escribir de la siguiente forma

$$CB(\theta_1, \theta_2, d) = CB_1(\theta_1) + CB_2(\theta_1, \theta_2) + CB_3(\theta_1, \theta_2, d)$$

donde

$$CB_1(\theta_1) = \int A_1(\theta_1, r, \varphi)\delta(z)B(r, \varphi, z)drd\varphi dz \quad (6.48)$$

$$CB_2(\theta_1, \theta_2) = \int A_2(\theta_1, \theta_2, r, \varphi)\delta(z)B(r, \varphi, z)drd\varphi dz \quad (6.49)$$

$$CB_3(\theta_1, \theta_2, d) = \int A_3(\theta_1, \theta_2, d, r, \varphi, z)B(r, \varphi, z)drd\varphi dz \quad (6.50)$$

En este punto del método se van a considerar, por un lado, las expresiones correspondientes (?? y ??) a las dos articulaciones de revolución y, por otro, la asociada a la prismática (??). Entonces hay que encontrar unas nuevas funciones A'_1 y A'_2 definidas de $C' \times W'$ en R y A'_3 definida de $C' \times W''$ en R , de tal forma que no dependan de alguno de los parámetros que definen una configuración. Con este objetivo, hay que realizar una nueva elección de los sistemas de referencia para la definición de A'_1 y A'_2 y otra para A'_3 . La elección para A'_1 y A'_2 es la siguiente

- $F_{W'} \equiv F_{A_1}$ y $F_{A'_1} \equiv F_{A_1}$

$$\blacksquare F_{A'_2} \equiv F_{A_2}$$

y para A'_3

■ $F_{W''}$ se mueve solidariamente con el tercer elemento y tiene la misma orientación que F_{A_1}

$$\blacksquare F_{A'_3} \equiv F_{A_3}$$

La definición de estos sistemas se puede observar en la figura ???. En concreto, se han considerado dos sistemas de referencia para el espacio de trabajo ($F_{W'}$ y $F_{W''}$): uno asociado a las articulaciones de revolución y otro asociado a la articulación prismática. De esta manera se logra que respecto a $F_{W'}$ y $F_{W''}$ los elementos \mathbf{A}_1 y \mathbf{A}_3 permanezcan fijos.

Teniendo en cuenta estos nuevos sistemas, una configuración vendría especificada por la orientación de $F_{A'_1}$ y $F_{A'_2}$ con respecto de $F_{W'}$ y el desplazamiento de $F_{A'_3}$ con respecto a $F_{W''}$. Entonces, C' contiene un conjunto de configuraciones dado por $(0, \theta'_2, 0)$, siendo $\theta'_2 = \theta_2$. En W' y en W'' se consideran coordenadas cilíndricas, denotándolas, respectivamente, por (r', φ', z') y (r'', φ'', z'') .

Considerando la dependencia de los dos primeros elementos con los grados de libertad analizada previamente, las funciones A'_1, A'_2 de $C' \times W'$ en R estarían definidas como

$$A'_1(0, r', \varphi') = \begin{cases} 1 & \text{si } (r', \varphi') \in \mathbf{A}_1(0) \\ 0 & \text{si } (r', \varphi') \notin \mathbf{A}_1(0) \end{cases} \quad (6.51)$$

$$A'_2(0, \theta_2, r', \varphi') = \begin{cases} 1 & \text{si } (r', \varphi') \in \mathbf{A}_2(0, \theta_2) \\ 0 & \text{si } (r', \varphi') \notin \mathbf{A}_2(0, \theta_2) \end{cases} \quad (6.52)$$

De igual forma la función A'_3 de $C' \times W''$ en R vendría definida por

$$A'_3(0, \theta_2, 0, r'', \varphi'', z'') = \begin{cases} 1 & \text{si } (r'', \varphi'', z'') \in \mathbf{A}_3(0, \theta_2, 0) \\ 0 & \text{si } (r'', \varphi'', z'') \notin \mathbf{A}_3(0, \theta_2, 0) \end{cases} \quad (6.53)$$

En estas definiciones aparecen los siguientes conjuntos:

- $\mathbf{A}_1(0)$ es el subconjunto de puntos que representa al elemento \mathbf{A}_1 en la configuración 0 respecto de $F_{W'}$, ya que no depende de θ_2 ni de d .
- $\mathbf{A}_2(0, \theta_2)$ es el subconjunto de puntos de W que representa al elemento \mathbf{A}_2 en la configuración $(0, \theta_2)$ respecto de $F_{W'}$, independientemente de d .

- $\mathbf{A}_3(0, \theta_2, 0)$ es el subconjunto de puntos que representa al elemento \mathbf{A}_3 en la configuración $(0, \theta_2, 0)$ respecto de $F_{W''}$.

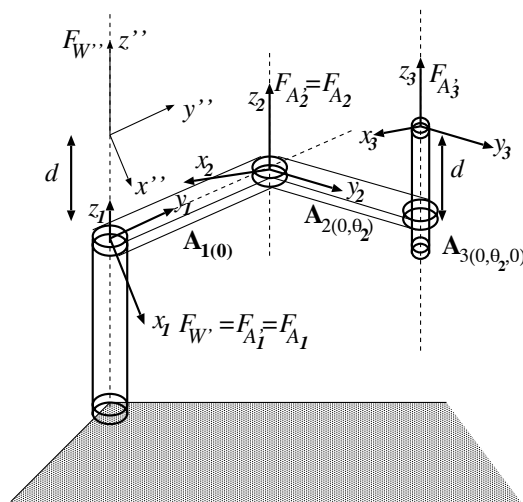


Figura 6.12: Definición de sistemas de referencia para A' del SCARA

Dado que se están considerando de forma separada los dos tipos de articulaciones, es necesario definir dos morfismos que lleven a cabo los cambios de sistemas de referencia : ϕ_1 de $C \times W$ en $C' \times W'$ y ϕ_2 de $C \times W$ en $C' \times W''$. El trabajar con coordenadas cilíndricas supone que exista una relación directa entre dos coordenadas del espacio de trabajo y dos del espacio de las configuraciones, concretamente, entre φ y θ_1 y entre z y d . Teniendo esta relación presente y las dependencias de los elementos con los grados de libertad, se elige el morfismo ϕ_1

$$\begin{aligned} \phi_1 : C \times W &\longrightarrow C' \times W' \\ \theta_1, \theta_2, d, r, \varphi, z &\longrightarrow \theta'_1, \theta'_2, d', r', \varphi', z' \end{aligned}$$

donde

$$\begin{aligned} r' &= r & \varphi' &= \varphi - \theta_1 \\ z' &= z & \theta'_1 &= 0 \\ \theta'_2 &= \theta_2 & d' &= d \end{aligned}$$

y el morfismo ϕ_2

$$\begin{aligned} \phi_2 : C \times W &\longrightarrow C' \times W'' \\ \theta_1, \theta_2, d, r, \varphi, z &\longrightarrow \theta'_1, \theta'_2, d', r'', \varphi'', z'' \end{aligned}$$

donde

$$\begin{aligned} r'' &= r & \varphi'' &= \varphi - \theta_1 \\ z'' &= z - d & \theta_1' &= 0 \\ \theta_2' &= \theta_2 & d' &= 0 \end{aligned}$$

Como $A_1 = (A_1' \circ \phi_1)$ y $A_2 = (A_2' \circ \phi_1)$ entonces

$$\begin{aligned} A_1(\theta_1, r, \varphi)\delta(z) &= A_1'(0, r, \varphi - \theta_1)\delta(z) \\ A_2(\theta_1, \theta_2, r, \varphi)\delta(z) &= A_2'(0, \theta_2, r, \varphi - \theta_1)\delta(z) \end{aligned}$$

Como $A_3 = (A_3' \circ \phi_2)$ entonces

$$A_3(\theta_1, \theta_2, d, r, \varphi, z) = A_3'(0, \theta_2, 0, r, \varphi - \theta_1, z - d)$$

Con las definiciones de funciones A (??, ?? y ??) y A' (??, ?? y ??), se tiene que

$$\begin{aligned} A_1'(0, r, \varphi - \theta_1)\delta(z) &= A_1(0, r, \varphi - \theta_1)\delta(z) \\ A_2'(0, \theta_2, r, \varphi - \theta_1)\delta(z) &= A_2(0, \theta_2, r, \varphi - \theta_1)\delta(z) \\ A_3'(0, \theta_2, 0, r, \varphi - \theta_1, z - d) &= A_3(0, \theta_2, 0, r, \varphi - \theta_1, z - d) \end{aligned}$$

y por tanto

$$A_1(\theta_1, r, \varphi)\delta(z) = A_1(0, r, \varphi - \theta_1)\delta(z) \quad (6.54)$$

$$A_2(\theta_1, \theta_2, r, \varphi)\delta(z) = A_2(0, \theta_2, r, \varphi - \theta_1)\delta(z) \quad (6.55)$$

$$A_3(\theta_1, \theta_2, d, r, \varphi, z) = A_3(0, \theta_2, 0, r, \varphi - \theta_1, z - d) \quad (6.56)$$

De estas expresiones se puede concluir que las funciones A_1 , A_2 y A_3 permanecen invariantes si se realiza un giro en la coordenada φ y un desplazamiento en la coordenada z , pasando el manipulador de adoptar la configuración (θ_1, θ_2, d) a adoptar $(0, \theta_2, 0)$.

Sustituyendo estas expresiones (??), (??) y (??) en (??), (??) y (??) se obtiene

$$CB_1(\theta_1) = \int A_{1(0)}(r, \varphi - \theta_1)B(r, \varphi, 0)drd\varphi$$

$$CB_2(\theta_1, \theta_2) = \int A_{2(0, \theta_2)}(r, \varphi - \theta_1)B(r, \varphi, 0)drd\varphi \quad (6.57)$$

$$CB_3(\theta_1, \theta_2, d) = \int A_{3(0, \theta_2, 0)}(r, \varphi - \theta_1, z - d)B(r, \varphi, z)drd\varphi dz \quad (6.58)$$

En las dos primeras expresiones la función B estaría definida en un plano $z = 0$, que es donde se ha considerado que se encuentran los dos primeros elementos. Para cada orientación θ_2 del segundo elemento \mathbf{A}_2 habría que calcular las dos integrales (??) y (??).

A partir de la expresión del producto de convolución de dos funciones definidas en R^3 sobre una o dos variables se obtiene

$$CB_1(\theta_1) = \int (\bar{A}_{1(0)} * B)_{\varphi}(r, \theta_1) dr \quad (6.59)$$

$$CB_2(\theta_1, \theta_2) = \int (\bar{A}_{2(0, \theta_2)} * B)_{\varphi}(r, \theta_1) dr \quad (6.60)$$

$$CB_3(\theta_1, \theta_2, d) = \int (\bar{A}_{3(0, \theta_2, 0)} * B)_{(\varphi, z)}(r, \theta_1, d) dr \quad (6.61)$$

donde

$$\begin{aligned} \bar{A}_{1(0)}(r, \theta_1 - \varphi) &= A_{1(0)}(r, \varphi - \theta_1) \\ \bar{A}_{2(0, \theta_2)}(r, \theta_1 - \varphi) &= A_{2(0, \theta_2)}(r, \varphi - \theta_1) \\ \bar{A}_{3(0, \theta_2, 0)}(r, \theta_1 - \varphi, d - z) &= A_{3(0, \theta_2, 0)}(r, \varphi - \theta_1, z - d) \end{aligned}$$

En las expresiones (??) y (??) sólo existe convolución en la variable θ_1 , mientras que en la expresión (??) la convolución es sobre las variables θ_1 y d . Además, para las dos últimas expresiones se debe realizar el cálculo de las integrales correspondientes para todos los valores de θ_2 en su intervalo de variación. Mientras que para las tres expresiones se debe efectuar la integral sobre r , tal que $r \in [-(l_2 + l_3), (l_2 + l_3)]$, donde l_2 y l_3 son las longitudes respectivas del segundo y tercer elemento.

Si se toman transformadas de Fourier en ambos miembros de estas expresiones y se aplica el teorema de convolución se obtiene

$$\begin{aligned} \mathcal{F}[CB_1(\theta_1)] &= \int \mathcal{F}[\bar{A}_{1(0)}(r, \theta_1)]_{\varphi} \mathcal{F}[B(r, \theta_1)]_{\varphi} dr \\ \mathcal{F}[CB_2(\theta_1, \theta_2)] &= \int \mathcal{F}[\bar{A}_{2(0, \theta_2)}(r, \theta_1)]_{\varphi} \mathcal{F}[B(r, \theta_1)]_{\varphi} dr \\ \mathcal{F}[CB_3(\theta_1, \theta_2, d)] &= \int \mathcal{F}[\bar{A}_{3(0, \theta_2, 0)}(r, \theta_1, d)]_{(\varphi, z)} \mathcal{F}[B(r, \theta_1, d)]_{(\varphi, z)} dr \end{aligned} \quad (6.62)$$

En las dos primeras se tienen transformadas de Fourier en una única dimensión, la coordenada angular, mientras que en la tercera la transformada es en dos dimensiones, la coordenada angular y la coordenada z .

La elección de un sistema de coordenadas cilíndricas para W , la representación $(\theta_1, \theta_2, d_3)$ de C y el tratamiento separado de las articulaciones prismática y de revolución han permitido definir dos morfismos que hace que las funciones A_1 , A_2 y A_3 definidas para el robot SCARA sean independientes de alguno o varios de sus parámetros, simplificándose el cálculo de CB correspondiente.

6.5. Robot Stanford

Si se reemplaza la tercera articulación de revolución del PUMA por una prismática o, lo que es lo mismo, si se añade un elemento con una articulación prismática a la cadena cinemática del manipulador reducido, se obtiene una estructura cinemática esférica (figura ??). El manipulador Stanford es el más representativo de esta estructura. Se van a denominar \mathbf{A}_1 , \mathbf{A}_2 y \mathbf{A}_3 a los tres elementos que componen la cadena, y θ_1 , θ_2 y d a sus tres grados de libertad.

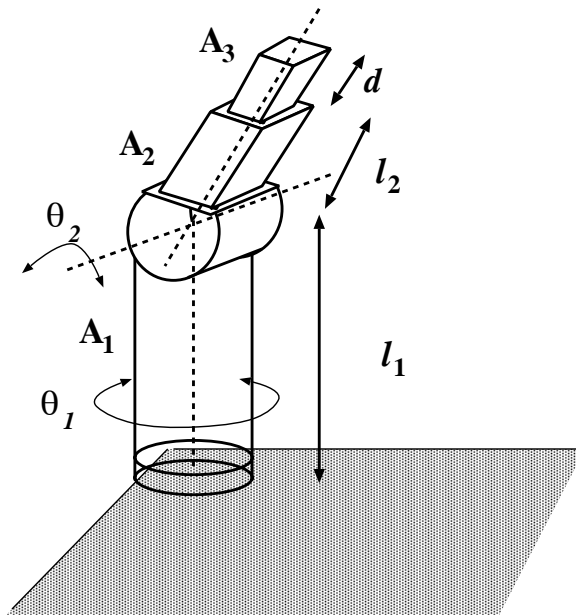


Figura 6.13: Elementos y articulaciones del manipulador Stanford

Los ejes de giro de las dos primeras articulaciones son perpendiculares entre sí, y, a la vez, perpendiculares al eje de desplazamiento de la tercera. Dado que los tres ejes intersectan en un punto, se han elegido los sistemas de referencia

(figura ??) siguiendo el siguiente procedimiento:

- Los orígenes de F_W y F_{A_1} se sitúan de forma que coincidan sobre el centro de la articulación que conecta los dos primeros elementos \mathbf{A}_1 y \mathbf{A}_2 . Los ejes z y z_1 , asociados con los sistemas F_W y F_{A_1} , se hacen coincidir con el eje de giro de la primera articulación. El resto de los ejes de ambos sistemas se eligen para formar un triedro, aunque entre x y x_1 (al igual que entre y e y_1) existe un desplazamiento angular dado por θ_1 .
- En el sistema F_{A_2} , el origen se sitúa coincidente con los de F_W y F_{A_1} . El eje y_2 se dispone a lo largo del segundo elemento \mathbf{A}_2 , y x_2 se hace coincidir con el eje de giro de la segunda articulación.
- El sistema F_{A_3} , asociado al tercer elemento \mathbf{A}_3 , es un sistema de referencia que se desplaza según lo hace \mathbf{A}_3 y su origen se ha colocado sobre el extremo superior de ese elemento. El eje y_3 se hace coincidir con el eje sobre el que se desplaza \mathbf{A}_3 . El eje x_3 se elige en la misma dirección que x_2 .

Una configuración de A vendría especificada por la orientación de F_{A_1} y F_{A_2} con respecto a F_W , denotadas respectivamente por θ_1 y θ_2 , y el desplazamiento de F_{A_3} con respecto a F_W , denominado d_1 siendo ($d_1 = d + l_2$). Su espacio de las configuraciones sería $S^1 \times S^1 \times R$. Como se ha comentado, cada configuración q vendría parametrizada por $(\theta_1, \theta_2, d_1) \in [0, 2\pi) \times [0, 2\pi) \times [l_2, l_2 + l_3]$ que corresponden respectivamente con los tres grados de libertad para las tres articulaciones. Con l_3 se denota la longitud máxima del tercer elemento y con l_2 la longitud del segundo elemento.

Dado que su estructura cinemática es esférica, parece lógico utilizar las coordenadas esféricas $(r, \varphi_1, \varphi_2)$ en el espacio de trabajo W . De esta forma se consigue que el desplazamiento d esté relacionado con r , mientras que el recorrido angular de θ_1 y θ_2 se relacione directamente con φ_1 y φ_2 , respectivamente. Las coordenadas elegidas para W tienen una correspondencia con los grados de libertad de este manipulador.

Con esta elección de coordenadas para W , es válida la definición de la función B (??) que se propuso para el manipulador reducido y, como consecuencia, para el PUMA.

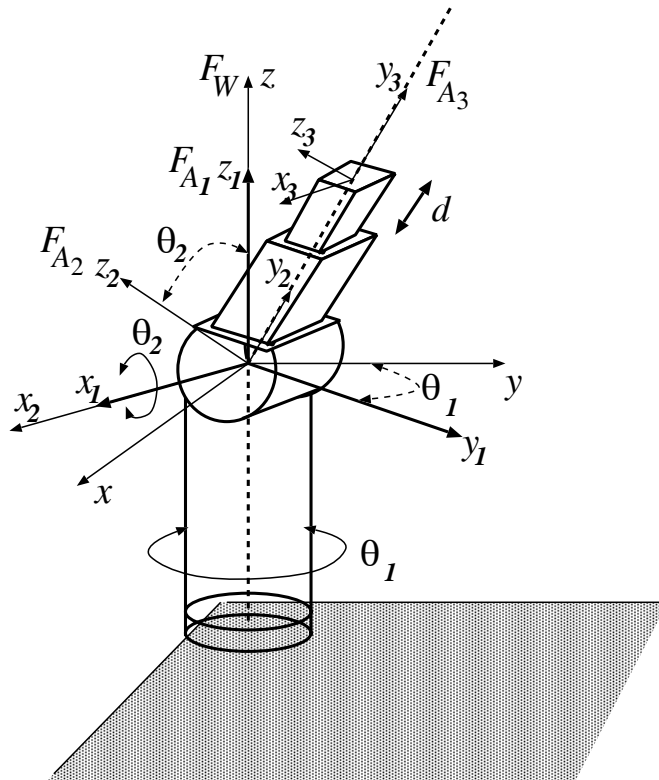


Figura 6.14: Definición de sistemas de referencia para el manipulador esférico

Una vez establecidos los espacios C y W , la definición de A tendría la misma forma que para el manipulador PUMA, modificando la dependencia con el giro θ_3 por el desplazamiento d_1 en el tercer elemento \mathbf{A}_3 .

$$A(\theta_1, \theta_2, d_1, r, \varphi_1, \varphi_2) = A_1(\theta_1, \theta_2, d_1, r, \varphi_1, \varphi_2) + A_2(\theta_1, \theta_2, d_1, r, \varphi_1, \varphi_2) + A_3(\theta_1, \theta_2, d_1, r, \varphi_1, \varphi_2)$$

A continuación se analizan las dependencias de los elementos con los grados de libertad:

- \mathbf{A}_1 no presenta dependencia con θ_2 ni con d .
- \mathbf{A}_2 no depende del desplazamiento d del tercer elemento.
- \mathbf{A}_3 sí depende de los tres grados de libertad.

Por tanto

$$A(\theta_1, \theta_2, d_1, r, \varphi_1, \varphi_2) = A_1(\theta_1, r, \varphi_1, \varphi_2) + A_2(\theta_1, \theta_2, r, \varphi_1, \varphi_2) + A_3(\theta_1, \theta_2, d_1, r, \varphi_1, \varphi_2) \quad (6.63)$$

Lógicamente, las funciones A_1 y A_2 estarían definidas como para el manipulador reducido o para el PUMA (?? y ??). La definición de A_3 sería

$$A_3(\theta_1, \theta_2, d_1, r, \varphi_1, \varphi_2) = \begin{cases} 1 & \text{si } (r, \varphi_1, \varphi_2) \in \mathbf{A}_3(\theta_1, \theta_2, d_1) \\ 0 & \text{si } (r, \varphi_1, \varphi_2) \notin \mathbf{A}_3(\theta_1, \theta_2, d_1) \end{cases} \quad (6.64)$$

donde $\mathbf{A}_3(\theta_1, \theta_2, d_1)$ es el subconjunto de W que representa al tercer elemento en la configuración $(\theta_1, \theta_2, d_1)$.

Entonces la definición de CB vendría dada por

$$CB(\theta_1, \theta_2, d_1) = \int A(\theta_1, \theta_2, d_1, r, \varphi_1, \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \quad (6.65)$$

Sustituyendo la expresión (??) en (??) se obtiene

$$CB(\theta_1, \theta_2, d_1) = CB_1(\theta_1) + CB_2(\theta_1, \theta_2) + CB_3(\theta_1, \theta_2, d_1)$$

donde

$$CB_1(\theta_1) = \int A_1(\theta_1, r, \varphi_1, \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \quad (6.66)$$

$$CB_2(\theta_1, \theta_2) = \int A_2(\theta_1, \theta_2, r, \varphi_1, \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \quad (6.67)$$

$$CB_3(\theta_1, \theta_2, d_1) = \int A_3(\theta_1, \theta_2, d_1, r, \varphi_1, \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \quad (6.68)$$

El procedimiento que se va seguir para reducir el tiempo de cálculo asociado a la resolución de estas integrales es análogo al realizado en el manipulador previo, ya que el esférico combina igualmente articulaciones de revolución y prismática. Por tanto, hay que encontrar unas nuevas funciones A'_1 y A'_2 definidas de $C' \times W'$ en R y A'_3 definida de $C' \times W''$ en R , de tal forma que las tres funciones no dependan de alguno de los parámetros que definen una configuración. Para lograr este objetivo, se deben elegir unos nuevos sistemas de referencia para estas tres funciones. Para las funciones A'_1 y A'_2 la elección es la siguiente:

- $F_{W'} \equiv F_{A_2}$
- $F_{A'_1} \equiv F_{A_2}$

$$\blacksquare F_{A'_2} \equiv F_{A_2}$$

Para la función A'_3

■ $F_{W''}$ se mueve solidariamente con el tercer elemento, su origen está situado en el extremo superior de este elemento y tiene la misma orientación que F_{A_2}

$$\blacksquare F_{A'_3} \equiv F_{A_3}$$

La definición de estos sistemas se puede observar en la figura ???. Se han definido dos sistemas de referencia para el espacio de trabajo: uno asociado a las articulaciones de revolución ($F_{W'}$) y otro para la prismática ($F_{W''}$). Los puntos pertenecientes al primer y tercer elemento del manipulador permanecen fijos respecto de estos sistemas.

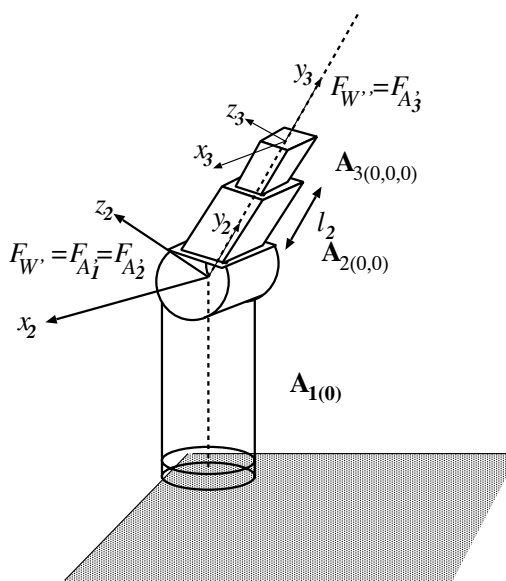
Considerando estos nuevos sistemas de referencia, una configuración vendría especificada por la orientación de $F_{A'_1}$ y $F_{A'_2}$ con respecto de $F_{W'}$ y el desplazamiento de $F_{A'_3}$ con respecto a $F_{W''}$. Entonces, C' contiene una única configuración dada por $q = (0, 0, 0)$. En W' y en W'' se eligen las coordenadas esféricas como coordenadas de trabajo, denotándolas, respectivamente, por $(r', \varphi'_1, \varphi'_2)$ y $(r'', \varphi''_1, \varphi''_2)$.

Las funciones A'_1 y A'_2 se definirían como en el manipulador reducido (?? y ??) o el PUMA; y la función A'_3 vendría definida por

$$A'_3(0, 0, 0, r'', \varphi''_1, \varphi''_2) = \begin{cases} 1 & \text{si } (r'', \varphi''_1, \varphi''_2) \in \mathbf{A}_{\mathbf{3}(0,0,0)} \\ 0 & \text{si } (r'', \varphi''_1, \varphi''_2) \notin \mathbf{A}_{\mathbf{3}(0,0,0)} \end{cases} \quad (6.69)$$

donde $\mathbf{A}_{\mathbf{3}(0,0,0)}$ es el subconjunto de puntos de W que representa al elemento $\mathbf{A}_{\mathbf{3}}$ en la configuración $(0, 0, 0)$ respecto de $F_{W''}$. Su representación, así como la de los subconjuntos $\mathbf{A}_{\mathbf{1}(0)}$ y $\mathbf{A}_{\mathbf{2}(0,0)}$, se puede observar en la figura ???.

La transformación de los sistemas de referencia definidos inicialmente a los nuevos es una traslación de coordenadas. Esto es debido al hecho de elegir las coordenadas esféricas como coordenadas para el espacio de trabajo W . De la misma forma que para el manipulador reducido o el PUMA, los giros θ_1 y θ_2 sobre los ejes z_1 y x_2 se convierten en traslaciones sobre las coordenadas angulares φ_1 y φ_2 . Además, para este manipulador se tiene que los desplazamientos sobre el tercer grado de libertad d se convierten en variaciones en la coordenada radial r .

Figura 6.15: Definición de sistemas de referencia para A' del Stanford

En este caso es necesario definir tres morfismos: $\phi_1 : C \times W \rightarrow C' \times W'$, $\phi_2 : C \times W \rightarrow C'' \times W''$ y $\phi_3 : C \times W \rightarrow C''' \times W'''$. Los morfismos ϕ_1 y ϕ_2 , que realizan la traslación en las coordenadas angulares, se definen de forma análoga a la utilizada en el manipulador reducido; el morfismo ϕ_1 es

$$\begin{aligned} \phi_1 : C \times W &\longrightarrow C' \times W' \\ \theta_1, \theta_2, d_1, r, \varphi_1, \varphi_2 &\longrightarrow \theta'_1, \theta'_2, d'_1, r', \varphi'_1, \varphi'_2 \end{aligned}$$

donde

$$\begin{aligned} r' &= r & \varphi'_1 &= \varphi_1 - \theta_1 \\ \varphi'_2 &= \varphi_2 & \theta'_1 &= 0 \\ \theta'_2 &= \theta_2 & d'_1 &= d_1 \end{aligned}$$

y el morfismo ϕ_2 es

$$\begin{aligned} \phi_2 : C \times W &\longrightarrow C'' \times W'' \\ \theta_1, \theta_2, d_1, r, \varphi_1, \varphi_2 &\longrightarrow \theta''_1, \theta''_2, d''_1, r'', \varphi''_1, \varphi''_2 \end{aligned}$$

donde

$$\begin{aligned} r'' &= r & \varphi''_1 &= \varphi_1 - \theta_1 \\ \varphi''_2 &= \varphi_2 - \theta_2 & \theta''_1 &= 0 \\ \theta''_2 &= 0 & d''_1 &= d_1 \end{aligned}$$

Por su parte el morfismo ϕ_3 se define como

$$\begin{aligned} C \times W &\longrightarrow C''' \times W''' \\ \theta_1, \theta_2, d_1, r, \varphi_1, \varphi_2 &\longrightarrow \theta_1''', \theta_2''', d_1''', r''', \varphi_1''', \varphi_2''' \end{aligned}$$

donde

$$\begin{aligned} r''' &= r - d_1 & \varphi_1''' &= \varphi_1 - \theta_1 \\ \varphi_2''' &= \varphi_2 - \theta_2 & \theta_1''' &= 0 \\ \theta_2''' &= 0 & d_1''' &= 0 \end{aligned}$$

Puesto que se cumple que $A_1 = (A'_1 \circ \phi_1)$ y $A_2 = (A'_2 \circ \phi_2)$ se obtiene que

$$\begin{aligned} A_1(\theta_1, r, \varphi_1, \varphi_2) &= A'_1(0, r, \varphi_1 - \theta_1, \varphi_2) \\ A_2(\theta_1, \theta_2, r, \varphi_1, \varphi_2) &= A'_2(0, 0, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \end{aligned}$$

y como $A_3 = (A'_3 \circ \phi_3)$, entonces

$$A_3(\theta_1, \theta_2, d_1, r, \varphi_1, \varphi_2) = A'_3(0, 0, 0, r - d_1, \varphi_1 - \theta_1, \varphi_2 - \theta_2)$$

A partir de las definiciones de las funciones A (??, ?? y ??) y A' (??, ?? y ??), se tiene que

$$\begin{aligned} A'_1(0, r, \varphi_1 - \theta_1, \varphi_2) &= A_1(0, r, \varphi_1 - \theta_1, \varphi_2) \\ A'_2(0, 0, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) &= A_2(0, 0, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \\ A'_3(0, 0, 0, r - d_1, \varphi_1 - \theta_1, \varphi_2 - \theta_2) &= A_3(0, 0, 0, r - d_1, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \end{aligned}$$

y por tanto

$$A_1(\theta_1, r, \varphi_1, \varphi_2) = A_1(0, r, \varphi_1 - \theta_1, \varphi_2) \quad (6.70)$$

$$A_2(\theta_1, \theta_2, r, \varphi_1, \varphi_2) = A_2(0, 0, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \quad (6.71)$$

$$A_3(\theta_1, \theta_2, d_1, r, \varphi_1, \varphi_2) = A_3(0, 0, 0, r - d_1, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \quad (6.72)$$

De estas tres expresiones se llega a que las funciones asociadas a los tres elementos de este manipulador permanecen invariantes si se realiza una traslación de coordenadas respecto de las variables esféricas $(r, \varphi_1, \varphi_2)$, pasando el robot de la configuración $(d_1, \theta_1, \theta_2)$ a la $(0, 0, 0)$. Hay que tener en cuenta que las traslaciones sobre las coordenadas esféricas φ_1 y φ_2 , elegidas para el espacio de trabajo, serían realmente giros si se hubiera optado por trabajar en coordenadas cartesianas.

Por notación se tiene que

$$\begin{aligned} A_{1(0)}(r, \varphi_1 - \theta_1, \varphi_2) &= A_1(0, r, \varphi_1 - \theta_1, \varphi_2) \\ A_{2(0,0)}(r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) &= A_2(0, 0, r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \\ A_{3(0,0,0)}(r - d_1, \varphi_1 - \theta_1, \varphi_2 - \theta_2) &= A_3(0, 0, 0, r - d_1, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \end{aligned}$$

Considerando las expresiones (??), (??) y (??) junto con las obtenidas previamente (??), (??) y (??) se verifica que

$$\begin{aligned} CB_1(\theta_1) &= \int A_{1(0)}(r, \varphi_1 - \theta_1, \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \\ CB_2(\theta_1, \theta_2) &= \int A_{2(0,0)}(r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \quad (6.73) \\ CB_3(\theta_1, \theta_2, d_1) &= \int A_{3(0,0,0)}(r - d_1, \varphi_1 - \theta_1, \varphi_2 - \theta_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \end{aligned}$$

Suponiendo que no existe colisión entre el elemento \mathbf{A}_1 del manipulador y algún obstáculo en W , entonces $CB_1 = 0$, y como consecuencia

$$CB(\theta_1, \theta_2, d_1) = CB_2(\theta_1, \theta_2) + CB_3(\theta_1, \theta_2, d_1)$$

La expresión (??) coincide con la obtenida (??) para el manipulador reducido en tres dimensiones. Para el manipulador Stanford aparece un término adicional que tiene en cuenta la tercera articulación prismática. Ya se comentó que su estructura cinemática se puede considerar como la composición de la del reducido y una tercera articulación prismática adicional. Además, la expresión para CB_2 es similar a la del PUMA, aunque el término para CB_3 es más simple, ya que el morfismo definido en este caso relaciona directamente los grados de libertad con las coordenadas de trabajo elegidas.

Considerando la convolución de dos funciones definidas en R^3 sobre dos de las variables se llega a

$$CB_2(\theta_1, \theta_2) = \int (\bar{A}_{2(0,0)} * B)_{(\varphi_1, \varphi_2)}(r, \theta_1, \theta_2) dr \quad (6.74)$$

y con la convolución de dos funciones definidas en R^3 sobre las tres variables se consigue

$$CB_3(\theta_1, \theta_2, d_1) = (\bar{A}_{3(0,0,0)} * B)_{(r, \varphi_1, \varphi_2)}(d_1, \theta_1, \theta_2) \quad (6.75)$$

donde

$$\begin{aligned} \bar{A}_{2(0,0)}(r, \theta_1 - \varphi_1, \theta_2 - \varphi_2) &= A_{2(0,0)}(r, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \\ \bar{A}_{3(0,0,0)}(d_1 - r, \theta_1 - \varphi_1, \theta_2 - \varphi_2) &= A_{3(0,0,0)}(r - d_1, \varphi_1 - \theta_1, \varphi_2 - \theta_2) \end{aligned}$$

Para la expresión (??) se produce convolución sobre dos de las variables θ_1 y θ_2 . Por tanto, si se toman en ella transformadas de Fourier en dos dimensiones se cumple que

$$\mathcal{F}[CB_2(\theta_1, \theta_2)] = \int \left(\mathcal{F}[\bar{A}_{2(0,0)}(r, \theta_1, \theta_2)]_{(\varphi_1, \varphi_2)} \mathcal{F}[B(r, \theta_1, \theta_2)]_{(\varphi_1, \varphi_2)} \right) dr \quad (6.76)$$

En (??) aparece el producto de convolución de dos funciones definidas en R^3 sobre las tres variables d_1 , θ_1 y θ_2 . Tomando transformadas de Fourier tridimensionales se tiene

$$\mathcal{F}[CB_3(\theta_1, \theta_2, d_1)] = \mathcal{F}[\bar{A}_{3(0,0,0)}(d_1, \theta_1, \theta_2)]_{(r, \varphi_1, \varphi_2)} \mathcal{F}[B(d_1, \theta_1, \theta_2)]_{(r, \varphi_1, \varphi_2)} \quad (6.77)$$

Particularizando un método general, basado en un formalismo matemático, se ha obtenido un conjunto de expresiones que simplifican el cálculo de obstáculos en el espacio de las configuraciones para un grupo determinado de estructuras robóticas. En el siguiente capítulo se va a presentar la forma de discretizar estas expresiones para llegar a proponer los algoritmos que permitan implementar y resolver este cálculo. Junto con ellos se analizará la complejidad computacional, intentando resaltar cómo ésta se reduce.

Capítulo 7

Algoritmos para Calcular los C-obstáculos

En los dos capítulos precedentes se han obtenido las expresiones particulares para el cálculo de los C-obstáculos, siguiendo el método matemático propuesto, tanto para robots móviles como para los manipuladores industriales más comunes. En este capítulo se van a proponer los algoritmos que permitan computar estas expresiones. El resultado de dichos algoritmos será una estructura donde estén representados los C-obstáculos. Esa estructura podrá ser utilizada, posteriormente, por el planificador o por el controlador. Con objeto de validar el método propuesto, los algoritmos se van a aplicar a cuatro situaciones reales: un disco, una plataforma rectangular, manipulador planar de revolución y un PUMA. Se comprobará que los resultados obtenidos para los tres primeros coinciden con los que aparecen en la bibliografía, mientras que para el último no existen prácticamente trabajos previos.

7.1. Discretización

Para computar las expresiones presentadas para el cálculo de los C-obstáculos, es necesario discretizar tanto el espacio de trabajo W como el C-espacio C . Esto lleva a que las funciones A y B estén definidas sobre espacios discretos y, por tanto, sea necesario proponer una nueva definición de CB para espacios

discretos. Este es el objetivo de esta sección.

7.1.1. Discretización de los espacios W y C

Sea x un punto de W y q una configuración de C , parametrizados, respectivamente, por $x = (x_1, x_2, \dots, x_n)$ y $q = (q_1, q_2, \dots, q_m)$. En general, n es 2 ó 3, mientras que m depende del número de grados de libertad del robot considerado. Sobre el intervalo de definición de cada una de estas coordenadas se distribuyen de manera uniforme un número finito de puntos (N). Por ejemplo, si una de las coordenadas de x o de q es una variable lineal $d \in [a, b]$, se distribuyen uniformemente N puntos, dando lugar a un vector cuya componente i -ésima sería

$$a + i \frac{b - a}{N}$$

y si la coordenada es angular $\theta \in [-\pi, \pi]$, sería

$$-\pi + i \frac{2\pi}{N}$$

Para simplificar la presentación del método, se supone que todas las coordenadas se discretizan con el mismo número de puntos. Realmente, en la implementación realizada no todas las coordenadas se discretizan con la misma resolución. Esta se debe fijar de forma que la representación de los obstáculos del robot en ambos espacios, W y C , sea correcta.

Se puede utilizar el índice $i \in D$, siendo $D = \{1, \dots, N\}$, como variable independiente para representar cada uno de los N puntos que se han distribuido sobre el intervalo de definición de cada coordenada de $x \in W$ o $q \in C$. Así, se obtiene una matriz de dimensión $\underbrace{N \times \dots \times N}_n$ para un espacio de trabajo de dimensión n y otra de dimensión $\underbrace{N \times \dots \times N}_m$ para el espacio de las configuraciones.

7.1.2. Discretización de las funciones A y B

Las funciones B , A y A' definidas sobre dominios discretos se denominarán, respectivamente, B^* , A^* y A'^* . Si se utiliza el índice como variable independiente entonces B^* , A^* y A'^* son funciones definidas sobre D , un subconjunto de los naturales, y, por tanto, se pueden representar como matrices. Debido a la propia

definición de B , A y A' , las correspondientes muestreadas dan como resultado matrices binarias.

Así, la función $B^* : \underbrace{D \times \dots \times D}_n \rightarrow R$ proporciona una matriz binaria que representa a los obstáculos en el espacio de trabajo. Un elemento de esta matriz B^* tomará un valor '1' si existe un obstáculo en la celda que representa dicho elemento, y si no el valor será '0'. De la misma forma, la función $A^* : \underbrace{D \times \dots \times D}_m \times \underbrace{D \times \dots \times D}_n \rightarrow R$ proporciona una matriz binaria que representa al robot en una determinada configuración q en el espacio de trabajo. Mientras que si se utiliza la función $A'^* : C' \times W' \rightarrow R$ se obtiene una matriz binaria que representa al robot en una configuración q' en W' . Cabe recordar que las funciones A' se introducían para simplificar el cálculo de $CB(q)$. Como se podrá comprobar con el siguiente ejemplo se puede reducir además las necesidades de memoria, ya que las matrices con las que se trabaja son de menor tamaño.

- **Ejemplo** Sea un robot con forma de disco que se mueve libremente en un espacio de trabajo $W = [a, b] \times [c, d]$. Un punto de W vendría parametrizado por $(x, y) \in [a, b] \times [c, d]$, y una configuración de C por $(x_r, y_r) \in [a, b] \times [c, d]$, que son las coordenadas del centro del móvil.

Si se discretiza W en una matriz $N \times N$, el elemento (x_i, y_j) o su índice (i, j) representaría a la celda espacial

$$\left[a + i \frac{b-a}{N}, a + (i+1) \frac{b-a}{N} \right] \times \left[c + j \frac{d-c}{N}, c + (j+1) \frac{d-c}{N} \right]$$

con $i, j \in D$, siendo $D = \{1, \dots, N\}$. De la misma manera se tendría si se discretiza C .

La función $B^* : W \rightarrow R$, definida sobre un dominio discreto, vendría dada por

$$B^*(x_i, y_j) = \begin{cases} 1 & \text{si } (x_i, y_j) \in \mathbf{B} \\ 0 & \text{si } (x_i, y_j) \notin \mathbf{B} \end{cases} \quad (7.1)$$

donde \mathbf{B} es el subconjunto de W que representa a los obstáculos. Utilizando el índice como variable independiente, la función $B^* : D \times D \rightarrow R$ se expresaría como

$$B^*(i, j) = \begin{cases} 1 & \text{si } (x_i, y_j) \in \mathbf{B} \\ 0 & \text{si } (x_i, y_j) \notin \mathbf{B} \end{cases} \quad (7.2)$$

Proporciona una matriz binaria de tamaño $N \times N$, cuyo elemento (i, j) toma valor '1' si hay un obstáculo en la celda (x_i, y_j) , y '0' en caso contrario.

Y la función $A^* : C \times W \rightarrow R$, siendo W y C discretos, vendría definida como

$$A^*(x_{r_k}, y_{r_l}, x_i, y_j) = \begin{cases} 1 & \text{si } (x_i, y_j) \in \mathbf{A}_{(x_{r_k}, y_{r_l})} \\ 0 & \text{si } (x_i, y_j) \notin \mathbf{A}_{(x_{r_k}, y_{r_l})} \end{cases} \quad (7.3)$$

donde $i, j, k, l \in D$, siendo $D = \{1, \dots, N\}$; $\mathbf{A}_{(x_{r_k}, y_{r_l})}$ es el subconjunto de puntos que representa al robot en la configuración (x_{r_k}, y_{r_l}) . Si se utilizan los índices, la definición de $A^* : \underbrace{D \times \dots \times D}_4 \rightarrow R$ sería

$$A^*(k, l, i, j) = \begin{cases} 1 & \text{si } (x_i, y_j) \in \mathbf{A}_{(x_{r_k}, y_{r_l})} \\ 0 & \text{si } (x_i, y_j) \notin \mathbf{A}_{(x_{r_k}, y_{r_l})} \end{cases} \quad (7.4)$$

y se obtiene una matriz binaria de tamaño $\underbrace{N \times \dots \times N}_4$. Su elemento (k, l, i, j) vale '1' si el robot en la configuración (x_{r_k}, y_{r_l}) ocupa la celda (x_i, y_j) del espacio de trabajo, y '0' en caso contrario. Sin embargo, en la expresión (??) se llegaba a

$$\begin{aligned} A(x_r, y_r, x, y) &= A(0, 0, x - x_r, y - y_r) \\ &= A_{(0,0)}(x - x_r, y - y_r) \end{aligned}$$

Luego, en lugar de trabajar con la matriz completa A^* de tamaño $\underbrace{N \times \dots \times N}_4$ es posible utilizar únicamente la submatriz o capa correspondiente a $A^*(0, 0, x_i - x_{r_k}, y_j - y_{r_l})$ correspondiente a la configuración de referencia $(0, 0)$, aunque los índices i, j sufren un desplazamiento dado por k, l . Concretamente, se emplea la matriz $A_{(0,0)}^*$ definida como

$$A_{(0,0)}^*(i - k, j - l) = \begin{cases} 1 & \text{si } (x_i - x_{r_k}, y_j - y_{r_l}) \in \mathbf{A}_{(0,0)} \\ 0 & \text{si } (x_i - x_{r_k}, y_j - y_{r_l}) \notin \mathbf{A}_{(0,0)} \end{cases} \quad (7.5)$$

que al ser de menor tamaño, $N \times N$, se logra reducir considerablemente las necesidades de memoria en la etapa de cálculo.

7.1.3. Discretización de la función CB

De la misma forma que para las funciones previas, se denomina CB^* a la función CB que toma valores sobre el C-espacio discreto, y estaría definida por

$$CB^*(q_j) = \sum_{i=0}^{N-1} A^*(q_j, x_i) B^*(x_i) \quad \forall q_j \in C, \quad \forall x_i \in W \quad (7.6)$$

A partir de ella, la región del C-obstáculo \mathbf{CB}_f , que es el subconjunto puntos del C-espacio discreto donde se proyecta el obstáculo \mathbf{B} , se define como

$$\mathbf{CB}_f = \{q_j \in C / CB^*(q_j) > 0\} \quad (7.7)$$

Por tanto, se cumple que un punto del C-espacio discreto es libre si y sólo si $CB^*(q_j) = 0$.

Si se trabaja con los índices el C-espacio se puede representar por una matriz m -dimensional (o por m submatrices $(m-1)$ -dimensionales). Así, la función $CB^* : \underbrace{D \times \dots \times D}_m \rightarrow R$ proporciona una matriz que representa a los obstáculos en el espacio de trabajo. Un determinado elemento de esta matriz CB^* tomará un valor no nulo si el robot en la configuración que esa celda representa colisiona con los obstáculos; o un valor '0' si es libre.

- **Ejemplo** Para el móvil con forma de disco analizado previamente la función $CB^* : C \rightarrow R$, definida sobre un dominio discreto, vendría dada por

$$CB^*(x_{r_k}, y_{r_l}) = \sum_{i,j=0}^{N-1} A_{(0,0)}^*(x_i - x_{r_k}, y_j - y_{r_l}) B^*(x_i, y_j)$$

de tal forma que

$$CB^*(x_{r_k}, y_{r_l}) = \begin{cases} P & \text{si } (x_{r_k}, y_{r_l}) \in \mathbf{CB}_f \\ 0 & \text{si } (x_{r_k}, y_{r_l}) \notin \mathbf{CB}_f \end{cases} \quad (7.8)$$

donde $P \in R$ y \mathbf{CB}_f es el subconjunto de C que representa a los C-obstáculos. Utilizando el índice como variable independiente, la función $CB^* : D \times D \rightarrow R$ se expresaría como

$$CB^*(k, l) = \sum_{i,j=0}^{N-1} A_{(0,0)}^*(i-k, j-l) B^*(i, j)$$

cumpliéndose que

$$CB^*(k, l) = \begin{cases} P & \text{si } (x_{r_k}, y_{r_l}) \in \mathbf{CB}_f \\ 0 & \text{si } (x_{r_k}, y_{r_l}) \notin \mathbf{CB}_f \end{cases} \quad (7.9)$$

Si se toma P como valor umbral se obtiene una matriz binaria de dimensión $N \times N$, cuyo elemento (k, l) toma valor '1' si hay un C-obstáculo en la celda (x_{rk}, y_{rl}) , y '0' en caso contrario.

7.2. Algoritmos

En los dos capítulos previos se obtenían las expresiones de $CB(q)$ para robots móviles y articulados, y se observaba cómo en dichas expresiones aparecía el producto de convolución de A y B sobre unas ciertas variables. De la misma manera, en la expresión de $CB^*(q_j)$ particularizada para los robots analizados, surgirá la convolución discreta de A^* y B^* sobre dichas variables discretizadas. Utilizando el teorema de convolución para dominios discretos, se puede calcular la Transformada de Fourier Discreta (DFT) de $CB^*(q_j)$ como el producto punto a punto de la DFT de A^* y la de B^* , efectuando las transformadas sobre una o varias variables. La dimensión de la Transformada Discreta de Fourier y las variables concretas sobre las que se efectúa depende del tipo de robot considerado. El objetivo de las siguientes secciones es obtener la expresión de $CB^*(q_j)$ para cada estructura robótica estudiada en los dos capítulos anteriores y, de ella, el algoritmo que permite calcular la matriz binaria que representa al espacio de las configuraciones.

Si se implementa directamente la convolución de las matrices binarias A^* y B^* , la complejidad computacional del algoritmo es $O(N^{d^2})$, siendo N la resolución de la discretización utilizada y d la dimensión de la convolución.

Otra opción sería aplicar el teorema de convolución y utilizar el algoritmo de la Transformada Rápida de Fourier (FFT), cuya complejidad es $O(N^d \log N)$. Así, se conseguiría reducir la carga asociada a la ejecución de los algoritmos.

Para aplicar el algoritmo de FFT las funciones deben de ser periódicas. Sin embargo, las matrices binarias implicadas no son cíclicas. Una forma de solucionar este problema, [?], es colocar un '1' en todos los elementos del contorno de la matriz binaria que representa al espacio de trabajo. Así, además de conseguir funciones cíclicas, se asegura que el robot no pueda sobrepasar los límites de W .

7.2.1. Robot Circular Móvil en el Plano

En la sección previa, se llegó a la expresión

$$CB^*(x_{r_k}, y_{r_l}) = \sum_{i,j=0}^{N-1} A_{(0,0)}^*(x_{r_k} - x_i, y_{r_l} - y_j) B^*(x_i, y_j)$$

Si se realiza el siguiente cambio en la expresión previa

$$\bar{A}_{(0,0)}^*(x_i - x_{r_k}, y_j - y_{r_l}) = A_{(0,0)}^*(x_{r_k} - x_i, y_{r_l} - y_j)$$

entonces aparece el producto de convolución discreto sobre las dos coordenadas entre las funciones que describen, respectivamente, al robot en la configuración $(0, 0)$ y a los obstáculos en el espacio de trabajo.

$$CB^*(x_{r_k}, y_{r_l}) = (\bar{A}_{(0,0)}^* * B^*)(x_{r_k}, y_{r_l}) \quad (7.10)$$

Con el teorema de convolución en su versión discreta la expresión anterior se podría calcular como

$$DFT [CB^*(x_{r_k}, y_{r_l})] = DFT [\bar{A}_{(0,0)}^*(x_{r_k}, y_{r_l})] DFT [B^*(x_{r_k}, y_{r_l})] \quad (7.11)$$

que son transformadas discretas en dos dimensiones.

Para su implementación, habría que calcular la FFT bidimensional de las matrices binarias $\bar{A}_{(0,0)}^*$ y B^* y multiplicar ambas transformadas. Calculando la transformada inversa IFFT de este resultado se obtendría la matriz donde están representados los C-obstáculos. Concretamente, el algoritmo que se propone es el siguiente

Construir la matriz binaria $B^*(k, l)$, poniendo un '1' en sus límites

Calcular $\mathcal{F}[B^*(k, l)]$ en 2 dimensiones

Construir la matriz binaria $\bar{A}_{(0,0)}^*(k, l)$

Calcular $\mathcal{F}[\bar{A}_{(0,0)}^*(k, l)]$

Obtener $P = \mathcal{F}[B^*] \cdot \mathcal{F}[\bar{A}_{(0,0)}^*]$

Hacer $IP(k, l) = \mathcal{F}^{-1}[P]$ la FFT inversa

Asignar $CB^*(k, l) = 1$ si y sólo si $|IP(k, l)| > 0$

Se supone que N es la resolución de la discretización, tanto para las coordenadas del espacio de trabajo W como para las del C -espacio. Con el algoritmo

se obtiene una matriz binaria B^* de tamaño $N \times N$, otra $\bar{A}_{(0,0)}^*$ de tamaño $N \times N$ y, como resultado final del algoritmo, una matriz binaria CB^* de tamaño $N \times N$. Con estas consideraciones y al trabajar con FFT en dos dimensiones, la complejidad computacional del algoritmo es $O(N^2 \cdot \log(N))$.

7.2.2. Plataforma Bidimensional Móvil en el Plano

Sea una plataforma móvil, con cualquier forma geométrica, que puede trasladarse y rotar libremente sobre un espacio de trabajo $W \subset R^2$. Partiendo de la expresión obtenida para el cálculo de $CB(q)$ de este robot

$$CB(x_r, y_r, \theta_r) = \int \bar{A}_{(0,0,\theta_r)}(x_r - x, y_r - y)B(x, y)dxdy$$

y siguiendo un proceso de discretización similar al analizado para el robot ejemplo se obtiene que

$$CB^*(x_{r_k}, y_{r_l}, \theta_{r_m}) = \sum_{i,j=0}^{N-1} \bar{A}_{(0,0,\theta_{r_m})}^*(x_{r_k} - x_i, y_{r_l} - y_j)B^*(x_i, y_j)$$

donde la orientación θ_r del robot también ha sido discretizada. Para cada orientación θ_{r_m} aparece el producto de convolución discreto sobre las dos variables lineales de las funciones que describen al robot en la configuración $(0, 0, \theta_{r_m})$ y a los obstáculos en el espacio de trabajo.

$$CB^*(x_{r_k}, y_{r_l}, \theta_{r_m}) = (\bar{A}_{(0,0,\theta_{r_m})}^* * B^*)(x_{r_k}, y_{r_l}) \quad (7.12)$$

Como sobre la variable θ_{r_m} no se produce convolución, entonces se puede distribuir sobre su intervalo de variación $\theta_r \in [0, 2\pi]$ un número de puntos diferente (M) al utilizado para las variables lineales (N). De esta forma, CB^* se representa mediante M matrices de tamaño $N \times N$: una matriz para cada orientación discreta del robot. Esta técnica se denomina *slicing* y a cada una de las matrices se la denomina *slice*.

Utilizando el teorema de convolución discreta se obtendría

$$DFT [CB^*(x_{r_k}, y_{r_l}, \theta_{r_m})] = DFT \left[\bar{A}_{(0,0,\theta_{r_m})}^*(x_{r_k}, y_{r_l}) \right] DFT [B^*(x_{r_k}, y_{r_l})] \quad (7.13)$$

que son transformadas discretas en dos dimensiones. Cada *slice* se obtiene multiplicando punto a punto las dos matrices bidimensionales que se obtienen al

calcular la FFT bidimensional de \bar{A}^* y de B^* y, considerando posteriormente, la transformada inversa de dicho producto. El algoritmo propuesto aparece a continuación

Construir la matriz binaria $B^*(k, l)$, poniendo un '1' en sus límites

Calcular $\mathcal{F}[B^*(k, l)]$ en 2 dimensiones

Para cada orientación θ_{r_m}

Construir la matriz binaria $\bar{A}_{(0,0,\theta_{r_m})}^*(k, l)$

Calcular $\mathcal{F}[\bar{A}_{(0,0,\theta_{r_m})}^*]$

Obtener $P = \mathcal{F}[B^*] \cdot \mathcal{F}[\bar{A}_{(0,0,\theta_{r_m})}^*]$

Hacer $IP(k, l) = \mathcal{F}^{-1}[P]$ la FFT inversa

Asignar $CB^*(k, l, m) = 1$ si y sólo si $|IP(k, l)| > 0$

Suponiendo que las variables lineales se discretizan con una resolución N mientras que la variable angular θ_r con una resolución M , se dispone de una matriz binaria B^* de tamaño $N \times N$, una matriz binaria \bar{A}^* de tamaño $N \times N$ para cada orientación del robot y, como resultado final del algoritmo, M matrices de dimensión $N \times N$. Se utiliza la técnica de *slicing* sobre el grado de libertad, θ_r , donde no se produce convolución. Como las FFT son en dos dimensiones la complejidad computacional del algoritmo es $O(M \cdot N^2 \cdot \log(N))$.

7.2.3. Plataforma Móvil Tridimensional sobre un Plano

En este apartado se va a considerar una plataforma tridimensional que se mueve sobre un plano $z = 0$. No se impone ninguna clase de limitación en cuanto a su forma geométrica. Para este tipo de móvil se había obtenido previamente la expresión ?? para la función $CB(q)$

$$CB(x_r, y_r, \theta_r) = \int \bar{A}_{(0,0,\theta_r)}(x_r - x, y_r - y, z) B(x, y, z) dx dy dz$$

Si se aplica el proceso de discretización propuesto anteriormente se llega a que

$$CB^*(x_{r_k}, y_{r_l}, \theta_{r_m}) = \sum_{i,j,n=0}^{N-1} \bar{A}_{(0,0,\theta_{r_m})}^*(x_{r_k} - x_i, y_{r_l} - y_j, z_n) B^*(x_i, y_j, z_n)$$

El planteamiento es similar al utilizado para la plataforma bidimensional. Sin embargo, en este caso se debe efectuar el sumatorio sobre tres variables discretas, que son las dimensiones del espacio de trabajo. Para cada capa o *slice* θ_{r_m} se calcula un CB^* determinado. Para cada una de ellas aparece el producto de convolución sobre las dos variables del plano.

$$CB^*(x_{r_k}, y_{r_l}, \theta_{r_m}) = \sum_{n=0}^{N-1} (\bar{A}_{(0,0,\theta_{r_m})}^* * B^*)_{(x_i, y_j)}(x_{r_k}, y_{r_l}, z_n)$$

Si se aplica a esta expresión el teorema de convolución discreto, entonces $\mathcal{DFT}[CB^*(x_{r_k}, y_{r_l}, \theta_{r_m})]$ se puede calcular como

$$\mathcal{DFT}[CB^*] = \sum_{n=0}^{N-1} \mathcal{DFT} \left[\bar{A}_{(0,0,\theta_{r_m})}^*(x_{r_k}, y_{r_l}, z_n) \right]_{(x_i, y_j)} \mathcal{DFT} [B^*(x_r, y_r, z_n)]_{(x_i, y_j)}$$

donde las transformadas discretas de Fourier se calculan sobre las coordenadas del plano (x, y) . Para cada orientación θ_{r_m} se obtendría una matriz de tamaño $N \times N$, al igual que para la plataforma bidimensional. En total se tendrían M matrices, siendo M la resolución con la que se discretiza θ_{r_m} . Cada una de ellas representa a $\mathcal{DF}[CB^*]$ para cada orientación del robot. Cada matriz $\mathcal{DF}[CB^*]$ se calcula con el producto punto a punto de las transformadas de $\bar{A}_{(0,0,\theta_{r_m})}^*$ y B^* para cada plano $z_n = \text{constante}$, desde $z_n = 0$ hasta $z_n = \text{altura} - \text{movil}$ y, después, se suman dichos productos. Por tanto, las M matrices que representan al C-espacio se pueden obtener con el siguiente algoritmo

Para cada plano z_n

Construir la matriz binaria $B^*(k, l)$, poniendo un '1' en sus límites

Calcular $\mathcal{F}[B^*(k, l)]$ en 2 dimensiones

Para cada valor de la orientación θ_{r_m}

$PT = 0$

Para cada coordenada z_n

Construir $\bar{A}_{(0,0,\theta_{r_m})}^*(k, l)$

Calcular $\mathcal{F}[\bar{A}_{(0,0,\theta_{r_m})}^*]$

Obtener $P = \mathcal{F}[B^*] \cdot \mathcal{F}[\bar{A}_{(0,0,\theta_r)}^*]$

Acumular P en PT

Hacer $IP(k, l) = \mathcal{F}^{-1}[PT]$ la FFT inversa

Asignar $CB(k, l) = 1$ si y sólo si $|IP(k, l)| > 0$

Se supone que las variables lineales se discretizan con la misma resolución N , aunque para la variable z se podría considerar otra diferente, y la angular θ_r se discretiza con una resolución M . En este caso es necesario construir N matrices binarias B^* de tamaño $N \times N$ y, para cada orientación del robot, otras N matrices binarias \bar{A}^* de tamaño $N \times N$. El resultado del algoritmo son M matrices binarias CB^* de tamaño $N \times N$. Como se realizan $M \times N$ transformadas de Fourier en dos dimensiones de matrices de tamaño $N \times N$, entonces la complejidad computacional del algoritmo es $O(M \cdot N \cdot N^2 \cdot \log(N))$.

7.2.4. Manipulador planar de revolución

En esta sección se va a considerar un robot articulado con dos grados de libertad $(\theta_1, \theta_2) \in [0, 2\pi]$, que se mueve en R^2 . Como coordenadas de trabajo se eligieron (r, φ) que pueden variar en el intervalo $[0, r_{max}] \times [0, 2\pi]$. Al discretizar estas variables con una determinada resolución, las expresiones

$$\begin{aligned} CB_1(\theta_1) &= \int \bar{A}_{1(0)}(r, \theta_1 - \varphi) B(r, \varphi) dr d\varphi \\ CB_2(\theta_1, \theta_2) &= \int \bar{A}_{2(0, \theta_2)}(r, \theta_1 - \varphi) B(r, \varphi) dr d\varphi \end{aligned}$$

se transforman en

$$\begin{aligned} CB_1^*(\theta_{1_l}) &= \sum_{i,j=0}^{N-1} \bar{A}_{1(0)}^*(r_i, \theta_{1_l} - \varphi_j) B^*(r_i, \varphi_j) \\ CB_2^*(\theta_{1_l}, \theta_{2_m}) &= \sum_{i,j=0}^{N-1} \bar{A}_{2(0, \theta_{2_m})}^*(r_i, \theta_{1_l} - \varphi_j) B^*(r_i, \varphi_j) \end{aligned}$$

En este caso sólo, para ambas expresiones, se produce la convolución discreta sobre la variable θ_{1_l} .

$$\begin{aligned} CB_1^*(\theta_{1_l}) &= \sum_{i=0}^{N-1} (\bar{A}_{1(0)}^* * B^*)_{\varphi_j}(r_i, \theta_{1_l}) \\ CB_2^*(\theta_{1_l}, \theta_{2_m}) &= \sum_{i=0}^{N-1} (\bar{A}_{2(0, \theta_{2_m})}^* * B^*)_{\varphi_j}(r_i, \theta_{1_l}) \end{aligned}$$

En CB_2^* se debe calcular el producto de convolución sobre una variable, para cada orientación del segundo elemento θ_{2_m} .

Si se aplica el teorema de convolución discreto, las expresiones previas se pueden calcular como

$$\begin{aligned} DFT [CB_1^*(\theta_{1_l})] &= \sum_{i=0}^{N-1} DFT \left[\bar{A}_{1(0)}^*(r_i, \theta_{1_l}) \right]_{\varphi_j} DFT [B^*(r_i, \theta_{1_l})]_{\varphi_j} \\ DFT [CB_2^*(\theta_{1_l}, \theta_{2_m})] &= \sum_{i=0}^{N-1} DFT \left[\bar{A}_{2(0, \theta_{2_m})}^*(r_i, \theta_{1_l}) \right]_{\varphi_j} DFT [B^*(r_i, \theta_{1_l})]_{\varphi_j} \end{aligned}$$

donde se aparece la transformada discreta de Fourier en una sólo dimensión: la variable angular. A partir de las expresiones previas, el algoritmo propuesto se puede sintetizar tal y como aparece a continuación.

Construir la matriz binaria $B^*(i, j)$ en polares, poniendo un '1' en sus límites

Construir la matriz binaria $\bar{A}_{1(0)}^*(i, j)$ en coordenadas polares

$P_1T = 0$

Para cada radio r_i

Calcular $\mathcal{F} [B^*(i, j)]$ en una dimensión

Calcular $\mathcal{F} [\bar{A}_{1(0)}^*(i, j)]$ en una dimensión

Obtener $P_1 = \mathcal{F} [B^*(i, j)] \cdot \mathcal{F} [\bar{A}_{1(0)}^*(i, j)]$

Acumular en P_1T

Para cada orientación θ_{2_m}

Construir la matriz binaria $\bar{A}_{2(0, \theta_{2_m})}^*(i, j)$

Hacer $P_2T = P_1T$

Para cada radio r_i

Calcular $\mathcal{F} [\bar{A}_{2(0, \theta_{2_m})}^*(i, j)]$ en una dimensión

Calcular $P_1 = \mathcal{F} [B^*(i, j)] \cdot \mathcal{F} [\bar{A}_{2(0, \theta_{2_m})}^*(i, j)]$

Acumular en P_2T

Hacer $IP_2T = \mathcal{F}^{-1} [P_2T]$ la FFT inversa

Asignar $CB^*(l, m) = 1$ si y sólo si $|IP_2T| > 0$

En este proceso de discretización se utiliza la misma resolución N para las variables angulares, aunque no es necesario que sea la misma para θ_2 , y una resolución M para la variable radial. Se construyen: una matriz binaria B^* de tamaño $N \times M$, otra \bar{A}_1^* del mismo tamaño y N matrices \bar{A}_2^* de tamaño $N \times M$.

El algoritmo calcula una matriz binaria CB^* de tamaño $N \times N$. Hay que hacer notar que se utiliza la FFT en un sola dimensión y, como consecuencia, la complejidad computacional del algoritmo es $O(M \cdot N^2 \cdot \log(N))$.

7.2.5. Robot PUMA

Este manipulador dispone de tres articulaciones de revolución, cuyos grados de libertad asociados se han denominado previamente θ_1 , θ_2 y θ_3 con un recorrido angular $[0, 2\pi]$. Como coordenadas de trabajo para W , se seleccionaron las esféricas $(r, \varphi_1, \varphi_2)$, con $r \in [0, r_{max}]$ y $(\varphi_1, \varphi_2) \in [0, 2\pi]$.

Al distribuir un número determinado de puntos sobre el intervalo de variación de cada variable y siguiendo el proceso de discretización analizado, las expresiones calculadas en el capítulo anterior para cada uno de los tres términos de $CB(q)$

$$\begin{aligned} CB_1(\theta_1) &= \int \bar{A}_{1(0)}(r, \theta_1 - \varphi_1, \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \\ CB_2(\theta_1, \theta_2) &= \int \bar{A}_{2(0,0)}(r, \theta_1 - \varphi_1, \theta_2 - \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \\ CB_3(\theta_1, \theta_2, \theta_3) &= \int \bar{A}_{3(0,0,\theta_3)}(r, \theta_1 - \varphi_1, \theta_2 - \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \end{aligned}$$

se convierten en

$$\begin{aligned} CB_1^*(\theta_{1l}) &= \sum_{i,j,k=0}^{N-1} \bar{A}_{1(0)}^*(r_i, \theta_{1l} - \varphi_{1j}, \varphi_{2k}) B^*(r_i, \varphi_{1j}, \varphi_{2k}) \\ CB_2^*(\theta_{1l}, \theta_{2m}) &= \sum_{i,j,k=0}^{N-1} \bar{A}_{2(0,0)}^*(r_i, \theta_{1l} - \varphi_{1j}, \theta_{2m} - \varphi_{2k}) B^*(r_i, \varphi_{1j}, \varphi_{2k}) \\ CB_3^*(\theta_{1l}, \theta_{2m}, \theta_{3n}) &= \sum_{i,j,k=0}^{N-1} \bar{A}_{3(0,0,\theta_{3n})}^*(r_i, \theta_{1l} - \varphi_{1j}, \theta_{2m} - \varphi_{2k}) B^*(r_i, \varphi_{1j}, \varphi_{2k}) \end{aligned}$$

Comparando estas expresiones con la del producto de convolución discreto de funciones definidas en R^3 sobre una y dos variables se tiene que

$$\begin{aligned} CB_1^*(\theta_{1l}) &= \sum_{i,k=0}^{N-1} (\bar{A}_{1(0)}^* * B^*)_{\varphi_{1j}}(r_i, \theta_{1l}, \varphi_{2k}) \\ CB_2^*(\theta_{1l}, \theta_{2m}) &= \sum_{i=0}^{N-1} (\bar{A}_{2(0,0)}^* * B^*)_{(\varphi_{1j}, \varphi_{2k})}(r_i, \theta_{1l}, \theta_{2m}) \\ CB_3^*(\theta_{1l}, \theta_{2m}, \theta_{3n}) &= \sum_{i=0}^{N-1} (\bar{A}_{3(0,0,\theta_{3n})}^* * B^*)_{(\varphi_{1j}, \varphi_{2k})}(r_i, \theta_{1l}, \theta_{2m}) \end{aligned}$$

Se produce convolución sobre la variable θ_{1_l} en la primera expresión y las variables θ_{1_l} y θ_{2_m} en las otras dos expresiones. Luego al tomar transformadas de Fourier unidimensional y bidimensional, según el caso, sobre los dos miembros de las expresiones se llega a

$$DFT [CB_1^*] = \sum_{i,k=0}^{N-1} DFT \left[\bar{A}_{1(0)}^*(r_i, \theta_{1_l}, \varphi_{2_k}) \right]_{\varphi_{1_j}} DFT [B^*(r_i, \theta_{1_l}, \varphi_{2_k})]_{\varphi_{1_j}}$$

$$DFT [CB_2^*] = \sum_{i=0}^{N-1} DFT \left[\bar{A}_{2(0,0)}^*(r_i, \theta_{1_l}, \theta_{2_m}) \right]_{(\varphi_{1_j}, \varphi_{2_k})} DFT [B^*(r_i, \theta_{1_l}, \theta_{2_m})]_{(\varphi_{1_j}, \varphi_{2_k})}$$

$$DFT [CB_3^*] = \sum_{i=0}^{N-1} DFT \left[\bar{A}_{3(0,0,\theta_{3_n})}^*(r_i, \theta_{1_l}, \theta_{2_m}) \right]_{(\varphi_{1_j}, \varphi_{2_k})} DFT [B^*(r_i, \theta_{1_l}, \theta_{2_m})]_{(\varphi_{1_j}, \varphi_{2_k})}$$

Tomando como punto de partida estas tres expresiones, los obstáculos se pueden representar en el espacio de las configuraciones mediante el siguiente algoritmo. Por facilitar su comprensión, se ha dividido en tres partes, donde se implementa cada uno de los tres términos. Una versión optimizada del algoritmo se puede obtener aplicando las propiedades de las transformadas de Fourier.

Construir la matriz binaria $B^*(i, j, k)$ en esféricas, poniendo un '1' en sus límites.

Construir la matriz binaria $\bar{A}_{1(0)}^*(i, j, k)$ en coordenadas esféricas.

$$P_1 T = 0$$

Para cada radio r_i

$$P_1 T_i = 0$$

Para cada φ_{2_k}

Calcular $\mathcal{F} [B^*(i, j, k)]$ en una dimensión

Calcular $\mathcal{F} [\bar{A}_{1(0)}^*(i, j, k)]$ en una dimensión

Obtener $P_1 = \mathcal{F} [B^*(i, j, k)] \cdot \mathcal{F} [\bar{A}_{1(0)}^*(i, j, k)]$

Acumular P_1 en $P_1 T_i$ según φ_{2_k}

Acumular $P_1 T_i$ en $P_1 T$ según r_i

Construir la matriz binaria $\bar{A}_{2(0,0)}^*(i, j, k)$ en coordenadas esféricas

$$P_2 T = 0$$

Para cada radio r_i

Calcular $\mathcal{F} [B^*(i, j, k)]$ en dos dimensiones

Calcular $\mathcal{F} [\bar{A}_{2(0,0)}^*(i, j, k)]$ en dos dimensiones

Obtener $P_2 = \mathcal{F}[B^*(i, j, k)] \cdot \mathcal{F}[\bar{A}_{2(0,0)}^*(i, j, k)]$

Acumular P_2 en P_2T según r_i

Para cada fila φ_{2_k} de P_2T

$P_2T = P_2T + P_1T$ Para cada orientación θ_{3_n}

Construir la matriz binaria de $\bar{A}_{3(0,0,\theta_{3_n})}^*(i, j, k)$

$P_3T_n = 0$

Para cada radio r_i

Calcular $\mathcal{F}[\bar{A}_{3(0,0,\theta_{3_n})}^*(i, j, k)]$ en dos dimensiones

Obtener $P_3 = \mathcal{F}[B^*(i, j, k)] \cdot \mathcal{F}[\bar{A}_{3(0,0,\theta_{3_n})}^*(i, j, k)]$

Acumular P_3 en P_3T_n según r_i

$P_3T_n = P_3T_n + P_2T$

Hacer $IP_3T = \mathcal{F}^{-1}[P_3T_n]$ la FFT inversa en dos dimensiones

Asignar $CB^*(l, m, n) = 1$ si y sólo si $|IP_3T| > 0$

En el proceso de discretización se ha utilizado la misma resolución N para todas las variables, tanto del espacio de trabajo como del espacio de las configuraciones. Sin embargo, podría haber sido diferente, aunque sí es necesario que θ_1 y φ_1 se discreticen con la misma resolución N_1 , al igual que θ_2 y φ_2 con N_2 . Para θ_3 y r se puede tomar cualquier resolución, N_3 y M , respectivamente. Así, se deben construir: una matriz binaria B^* de tamaño $N_1 \times N_2 \times M$, otras dos \bar{A}_1^* y \bar{A}_2^* del mismo tamaño y N_3 matrices \bar{A}_3^* de tamaño $N_1 \times N_2 \times M$. El algoritmo calcula: un vector intermedio P_1T de tamaño $1 \times N_1$ correspondiente a $CB_1(\theta_1)$; una matriz P_2T de tamaño $N_1 \times N_2$ asociada con $CB_2(\theta_1, \theta_2)$; y N_3 matrices P_3T de tamaño $N_1 \times N_2$ correspondientes con $CB_3(\theta_1, \theta_2, \theta_3)$. El resultado del algoritmo es una matriz binaria CB^* de tamaño $N_1 \times N_2 \times N_3$.

Hay que hacer notar que se utilizan transformadas rápidas de Fourier en una dimensión para el término $CB_1(\theta_1)$ y en dos dimensiones para $CB_2(\theta_1, \theta_2)$ y $CB_3(\theta_1, \theta_2, \theta_3)$. Como consecuencia, la complejidad computacional del algoritmo es $O(N_3 \cdot N_3 \cdot M \cdot N^2 \log(N))$, donde se ha supuesto que $N_1 = N_2 = N$.

7.2.6. Robot SCARA

En el capítulo previo se denominaron θ_1 , θ_2 y d a los grados de libertad de este robot asociados, respectivamente, con las dos primeras articulaciones de revolución y con la tercera prismática. Su intervalo de variación es $(\theta_1, \theta_2, d) \in [0, 2\pi] \times [0, 2\pi] \times [0, l_3]$, donde l_3 es la longitud del tercer elemento. En el espacio de trabajo se eligieron, como coordenadas de trabajo, las cilíndricas (r, φ, z) . De esta forma, se lograba relacionar el recorrido angular de θ_1 con φ y el desplazamiento d con z . Con ello se llegaba a las siguientes expresiones (??) y (??) para $CB(q)$

$$\begin{aligned} CB_1(\theta_1) &= \int \bar{A}_{1(0)}(r, \theta_1 - \varphi) B(r, \varphi, 0) dr d\varphi \\ CB_2(\theta_1, \theta_2) &= \int \bar{A}_{2(0, \theta_2)}(r, \theta_1 - \varphi) B(r, \varphi, 0) dr d\varphi \\ CB_3(\theta_1, \theta_2, d) &= \int \bar{A}_{3(0, \theta_2, 0)}(r, \theta_1 - \varphi, d - z) B(r, \varphi, z) dr d\varphi dz \end{aligned}$$

En el proceso de discretización se distribuyen N puntos sobre el recorrido de todas variables, aunque sólo es necesario que sea el mismo número para φ y θ_1 y para d y z . Realizando dicho proceso se llega a

$$\begin{aligned} CB_1^*(\theta_{1_l}) &= \sum_{i,j=0}^{N-1} \bar{A}_{1(0)}^*(r_i, \theta_{1_l} - \varphi_j) B^*(r_i, \varphi_j, 0) \\ CB_2^*(\theta_{1_l}, \theta_{2_m}) &= \sum_{i,j=0}^{N-1} \bar{A}_{2(0, \theta_{2_m})}^*(r_i, \theta_{1_l} - \varphi_j) B^*(r_i, \varphi_j, 0) \\ CB_3^*(\theta_{1_l}, \theta_{2_m}, d_n) &= \sum_{i,j,k=0}^{N-1} \bar{A}_{3(0, \theta_{2_m}, 0)}^*(r_i, \theta_{1_l} - \varphi_j, d_i - z_k) B^*(r_i, \varphi_j, z_k) \end{aligned}$$

o a las expresiones

$$\begin{aligned} CB_1^*(\theta_{1_l}) &= \sum_{i=0}^{N-1} (\bar{A}_{1(0)}^* * B^*)_{\varphi_j}(r_i, \theta_{1_l}) \\ CB_2^*(\theta_{1_l}, \theta_{2_m}) &= \sum_{i=0}^{N-1} (\bar{A}_{2(0, \theta_{2_m})}^* * B^*)_{\varphi_j}(r_i, \theta_{1_l}) \\ CB_3^*(\theta_{1_l}, \theta_{2_m}, d_n) &= \sum_{i=0}^{N-1} (\bar{A}_{3(0, \theta_{2_m}, 0)}^* * B^*)_{(\varphi_j, z_k)}(r_i, \theta_{1_l}, d_n) \end{aligned}$$

Como ya se comentó en el capítulo anterior se produce convolución sobre las variables θ_{1_l} y d_n . Luego, si se toman transformadas discretas de Fourier en

una dimensión sobre las dos primeras expresiones y en dos dimensiones sobre la tercera expresión, se llega a:

$$DFT [CB_1^*] = \sum_{i=0}^{N-1} DFT \left[\bar{A}_{1(0)}^*(r_i, \theta_{1_l}) \right]_{\varphi_j} DFT [B^*(r_i, \theta_{1_l}, 0)]_{\varphi_j}$$

$$DFT [CB_2^*] = \sum_{i=0}^{N-1} DFT \left[\bar{A}_{2(0, \theta_{2_m})}^*(r_i, \theta_{1_l}) \right]_{\varphi_j} DFT [B^*(r_i, \theta_{1_l}, 0)]_{\varphi_j}$$

$$DFT [CB_3^*] = \sum_{i=0}^{N-1} DFT \left[\bar{A}_{3(0, \theta_{2_m}, 0)}^*(r_i, \theta_{1_l}, d_n) \right]_{(\varphi_j, z_k)} DFT [B^*(r_i, \theta_{1_l}, d_n)]_{(\varphi_j, z_k)}$$

Estas expresiones sirven de base para proponer el siguiente algoritmo para el cálculo del C-espacio.

Construir la matriz binaria $B_0^*(i, j)$ en cilíndricas, poniendo '1' en sus límites

Construir la matriz binaria $\bar{A}_{1(0)}^*(i, j)$ en cilíndricas sobre el plano $z = 0$

$$P_1 T = 0$$

Para cada radio r_i

Calcular $\mathcal{F} [B_0^*(i, j)]$ en una dimensión

Calcular $\mathcal{F} [\bar{A}_{1(0)}^*(i, j)]$ en una dimensión

Obtener $P_1 = \mathcal{F} [B_0^*(i, j)] \cdot \mathcal{F} [\bar{A}_{1(0)}^*(i, j)]$

Acumular P_1 en $P_1 T$ según r_i

Para cada orientación θ_{2_m}

Construir la matriz binaria $\bar{A}_{2(0, \theta_{2_m})}^*(i, j)$ en cilíndricas

$$P_2 T_m = 0$$

Para cada radio r_i

Calcular $\mathcal{F} [\bar{A}_{2(0, \theta_{2_m})}^*(i, j)]$ en una dimensión

Obtener $P_2 = \mathcal{F} [B_0^*(i, j)] \cdot \mathcal{F} [\bar{A}_{2(0, \theta_{2_m})}^*(i, j)]$

Acumular P_2 en $P_2 T_m$ según r_i

$$P_2 T_m = P_2 T_m + P_1 T$$

Construir la matriz binaria $B^*(i, j, k)$ en cilíndricas, poniendo '1' en sus límites

Para cada radio r_i

Calcular $\mathcal{F} [B^*(i, j, k)]$ en dos dimensiones

Para cada orientación θ_{2_m}

Construir la matriz binaria $\bar{A}_{3(0, \theta_{2_m}, 0)}^*(i, j, k)$ en cilíndricas

$$P_3T_m = 0$$

Para cada radio r_i

Calcular $\mathcal{F} \left[\bar{A}_{3(0,\theta_{2m},0)}^*(i,j,k) \right]$ en dos dimensiones

Obtener $P_3 = \mathcal{F} [B^*(i,j,k)] \cdot \mathcal{F} \left[\bar{A}_{3(0,\theta_{2m},0)}^*(i,j,k) \right]$

Acumular P_3 en P_3T_m según r_i

$$P_3T = P_3T + P_2$$

Para cada fila z_k de P_3T_m

$$P_3T_m = P_3T_m + P_2T_m$$

Hacer $IP_3T = \mathcal{F}^{-1} [P_3T_m]$ la FFT inversa en 2 dimensiones

Asignar $CB^*(l,m,n) = 1$ si y sólo si $|IP_3T| > 0$

Por facilitar la comprensión del algoritmo, se han implementado directamente los tres términos de CB sin intentar ningún tipo de optimización. De esta forma el algoritmo consta de tres módulos. En el primero, se construyen las matrices $\bar{A}_{1(0)}^*$ y B_0^* de tamaño $M \times N_1$, donde M y N_1 son, respectivamente, las resoluciones utilizadas para r , y para φ y θ_1 . En el segundo se construyen N_2 matrices $\bar{A}_{(0,\theta_{2m})}^*$ de tamaño $M \times N_1$, siendo N_2 la resolución para θ_{2m} . En el tercero se definen N_2 matrices $\bar{A}_{3(0,\theta_{2m})}^*$ de tamaño $M \times N_1 \times N_3$ y una matriz B^* de tamaño $M \times N_1 \times N_3$, donde N_3 es la resolución para z y d . El algoritmo calcula: un vector P_1T de tamaño $1 \times N_1$ asociado a CB_1 ; N_2 vectores P_2T de tamaño $1 \times N_1$ asociados a CB_2 ; y N_2 matrices P_3T de tamaño $N_1 \times N_3$ relacionados con CB_3 . El resultado del algoritmo es una matriz binaria, formada por N_2 slices de tamaño $N_1 \times N_3$, donde están representados los obstáculos en el espacio de las configuraciones.

La carga computacional del algoritmo está asociada principalmente con el tercer módulo, donde se calculan $N_2 \times M$ transformadas de Fourier en dos dimensiones. Por tanto, es $O(N_2 \cdot M \cdot N^2 \cdot \log(N))$, ya que se ha tomado $N_1 = N_3 = N$.

7.2.7. Robot de Stanford

Como se describió en el capítulo previo, este manipulador tiene dos articulaciones de revolución, con grados de libertad θ_1 y θ_2 , y la tercera prismática,

siendo d_1 su grado de libertad. Como coordenadas de trabajo se eligieron las esféricas $(r, \varphi_1, \varphi_2)$ y las expresiones que se obtuvieron para los tres sumandos de $CB(q)$ fueron

$$\begin{aligned} CB_1(\theta_1) &= \int \bar{A}_{1(0)}(r, \theta_1 - \varphi_1, \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \\ CB_2(\theta_1, \theta_2) &= \int \bar{A}_{2(0,0)}(r, \theta_1 - \varphi_1, \theta_2 - \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \\ CB_3(\theta_1, \theta_2, d_1) &= \int \bar{A}_{3(0,0,0)}(d_1 - r, \theta_1 - \varphi_1, \theta_2 - \varphi_2) B(r, \varphi_1, \varphi_2) dr d\varphi_1 d\varphi_2 \end{aligned}$$

Se distribuyen un número determinado de puntos sobre las coordenadas de trabajo elegidas y sobre el recorrido de cada grado de libertad $\theta_1, \theta_2 \in [0, 2\pi]$ y $d_1 \in [l_2, l_2 + l_3]$, donde l_2 es la longitud del segundo elemento y l_3 es la longitud máxima del tercer elemento. El número de puntos tiene que ser el mismo para θ_1 y φ_1 , para θ_2 y φ_2 , y para r y d_1 , respectivamente. No obstante, con el fin de simplificar se supondrá que es el mismo (N) para las seis variables. Teniendo esto en cuenta, con el proceso de discretización, se llega a

$$\begin{aligned} CB_1^*(\theta_{1_l}) &= \sum_{i,j,k=0}^{N-1} \bar{A}_{1(0)}^*(r_i, \theta_{1_l} - \varphi_{1_j}, \varphi_{2_k}) B^*(r_i, \varphi_{1_j}, \varphi_{2_k}) \\ CB_2^*(\theta_{1_l}, \theta_{2_m}) &= \sum_{i,j,k=0}^{N-1} \bar{A}_{2(0,0)}^*(r_i, \theta_{1_l} - \varphi_{1_j}, \theta_{2_m} - \varphi_{2_k}) B^*(r_i, \varphi_{1_j}, \varphi_{2_k}) \\ CB_3^*(\theta_{1_l}, \theta_{2_m}, d_{1_n}) &= \sum_{i,j,k=0}^{N-1} \bar{A}_{3(0,0,0)}^*(d_{1_n} - r_i, \theta_{1_l} - \varphi_{1_j}, \theta_{2_m} - \varphi_{2_k}) B^*(r_i, \varphi_{1_j}, \varphi_{2_k}) \end{aligned}$$

Considerando la definición del producto de convolución discreto de funciones definidas en R^3 sobre una, dos y tres variables, respectivamente, las expresiones anteriores se convierten en

$$\begin{aligned} CB_1^*(\theta_{1_l}) &= \sum_{i,k=0}^{N-1} (\bar{A}_{1(0)}^* * B^*)_{\varphi_{1_j}}(r_i, \theta_{1_l}, \varphi_{2_k}) \\ CB_2^*(\theta_{1_l}, \theta_{2_m}) &= \sum_{i=0}^{N-1} (\bar{A}_{2(0,0)}^* * B^*)_{(\varphi_{1_j}, \varphi_{2_k})}(r_i, \theta_{1_l}, \theta_{2_m}) \\ CB_3^*(\theta_{1_l}, \theta_{2_m}, d_{1_n}) &= (\bar{A}_{3(0,0,0)}^* * B^*)_{(r_i, \varphi_{1_j}, \varphi_{2_k})}(d_{1_n}, \theta_{1_l}, \theta_{2_m}) \end{aligned}$$

Tomando transformadas de Fourier de la dimensión apropiada en las expresiones previas y teniendo en cuenta el teorema de convolución, resulta

$$DFT[CB_1^*] = \sum_{i,k=0}^{N-1} DFT \left[\bar{A}_{1(0)}^*(r_i, \theta_{1_l}, \varphi_{2_k}) \right]_{\varphi_{1_j}} DFT[B^*(r_i, \theta_{1_l}, \varphi_{2_k})]_{\varphi_{1_j}}$$

$$DFT [CB_2^*] = \sum_{i=0}^{N-1} DFT \left[\bar{A}_{2(0,0)}^*(r_i, \theta_{1_l}, \theta_{2_m}) \right]_{(\varphi_{1_j}, \varphi_{2_k})} DFT [B^*(r_i, \theta_{1_l}, \theta_{2_m})]_{(\varphi_{1_j}, \varphi_{2_k})}$$

$$DFT [CB_3^*] = DFT \left[\bar{A}_{3(0,0,0)}^*(d_{1_n}, \theta_{1_l}, \theta_{2_m}) \right]_{(r_i, \varphi_{1_j}, \varphi_{2_k})} DFT [B^*(d_{1_n}, \theta_{1_l}, \theta_{2_m})]_{(r_i, \varphi_{1_j}, \varphi_{2_k})}$$

A continuación se propone un algoritmo para obtener CB , donde se calcula de forma separada cada uno de sus tres términos.

Construir $B^*(i, j, k)$ en esféricas, poniendo '1' en sus límites

Construir la matriz binaria $\bar{A}_{1(0)}^*(i, j, k)$ en coordenadas esféricas

$$P_1 T = 0$$

Para cada radio r_i

Para cada φ_{2_k}

Calcular $\mathcal{F} [B^*(i, j, k)]$ en una dimensión

Calcular $\mathcal{F} [\bar{A}_{1(0)}^*(i, j, k)]$ en una dimensión

Obtener $P_1 = \mathcal{F} [B^*(i, j, k)] \cdot \mathcal{F} [\bar{A}_{1(0)}^*(i, j, k)]$

Acumular P_1 en $P_1 T$ según r_i

Construir la matriz binaria $\bar{A}_{2(0,0)}^*(i, j, k)$ en coordenadas esféricas

$$P_2 T = 0$$

Para cada radio r_i

Calcular $\mathcal{F} [B^*(i, j, k)]$ en dos dimensiones

Calcular $\mathcal{F} [\bar{A}_{2(0,0)}^*(i, j, k)]$ en dos dimensiones

Obtener $P_2 = \mathcal{F} [B^*(i, j, k)] \cdot \mathcal{F} [\bar{A}_{2(0,0)}^*(i, j, k)]$

Acumular P_2 en $P_2 T$ según r_i

$$P_2 T = P_2 T + P_1 T$$

Construir la matriz binaria de $\bar{A}_{3(0,0,0)}^*(i, j, k)$

Calcular $\mathcal{F} [\bar{A}_{3(0,0,0)}^*(i, j, k)]$ en tres dimensiones

Calcular $\mathcal{F} [B^*(i, j, k)]$ en tres dimensiones

Obtener $P_3 T = \mathcal{F} [W^*(i, j, k)] \cdot \mathcal{F} [\bar{A}_{3(0,0,0)}^*(i, j, k)]$

Para cada fila r_i de $P_3 T$

$$P_3 T = P_3 T + P_2 T$$

Hacer $IP_3 T = \mathcal{F}^{-1} [P_3 T]$ la FFT inversa en tres dimensiones

Asignar $CB^*(l, m, n) = 1$ si y sólo si $|IP_3 T| > 0$

La complejidad computacional de este algoritmo es $O(N^3 \cdot \log(N))$, que está asociada al cálculo de CB_3^* , donde aparece FFT en tres dimensiones. Hay que tener en cuenta que los C-obstáculos generados por colisiones con los dos primeros elementos restringen fuertemente el movimiento de este manipulador. En situaciones reales estos choques no deben producirse, ya que sino el manipulador no podría moverse. Luego su cálculo podría obviarse. Además, la carga computacional que representan es mínima comparada con la asociada al cálculo de los choques entre el tercer elemento y los obstáculos.

7.3. Implementación y resultados

Uno de los objetivos más destacables de este trabajo de tesis es la proposición de los algoritmos para la representación de los C-obstáculos de los distintos robots. Se ha considerado conveniente la implementación de alguno de ellos con objeto de contrastar los resultados obtenidos con otros de la bibliografía. Así, se podrá validar la metodología propuesta. Hay que destacar que en algunos casos, entre los que se encuentra el PUMA, no existen resultados gráficos sobre la representación de los C-obstáculos.

En esta sección se van a comentar los aspectos más destacados de su implementación. Como se puede observar, en algunos algoritmos se pueden realizar ciertas optimizaciones y, así, reducir el tiempo de cálculo. Sin embargo, en su presentación se ha optado por la claridad para su fácil comprensión. También hay que destacar que la paralelización de los algoritmos es inmediata. Sin embargo, su realización queda fuera de nuestro trabajo.

Concretamente, los algoritmos se han implementado en una estación de cálculo Silicon Graphics Power Challenge XL con procesador MIPS R8000. Como herramienta software, se ha utilizado el lenguaje de programación C y el compilador MIPSpro versión 7.20. Aunque están disponibles multitud de librerías que permiten calcular la transformada rápida de Fourier de forma optimizada, se ha optado por una librería matemática propiedad del fabricante denominada CompLib.SgiMath, que es de uso generalizado en la comunidad científica.

Se han diseñado diversas funciones que, haciendo uso de las proporcionadas por el fabricante, calculan las transformadas de mapas de bits en una, dos y

tres dimensiones. En su implementación se ha considerado como parámetro la resolución N de la discretización. La importancia de este parámetro es doble, ya que, por una parte, define la exactitud de los resultados obtenidos y, por otra, marca el tiempo de cálculo y el consumo de memoria, como se ha mostrado en las secciones anteriores.

La reducción del tiempo de cálculo viene proporcionada por las posibilidades de paralelización del código, en cuanto a utilizar librerías paralelas en la realización de las transformadas de Fourier y en cuanto al carácter paralelizable de los algoritmos propuestos. Un aspecto que ha sido necesario resolver es el elevado consumo de memoria, que puede limitar la resolución utilizada. Una forma de evitar esta limitación es la utilización de matrices *sparse*. Su uso en este caso está plenamente justificado dado que en las situaciones reales los mapas de bits se encuentran vacíos en su mayor parte.

Se han examinado una serie de situaciones, cada una de ellas con objeto de mostrar algún aspecto relevante sobre la aplicación de los algoritmos. Por otra parte se han utilizado dos niveles de discretización $N = 64$ y $N = 128$, donde el primero se ha utilizado únicamente al robot tipo PUMA, pues en el resto de los casos los tiempos de cálculo han sido despreciables. Los tipos de robot, cuya aplicación se va a mostrar en este apartado son:

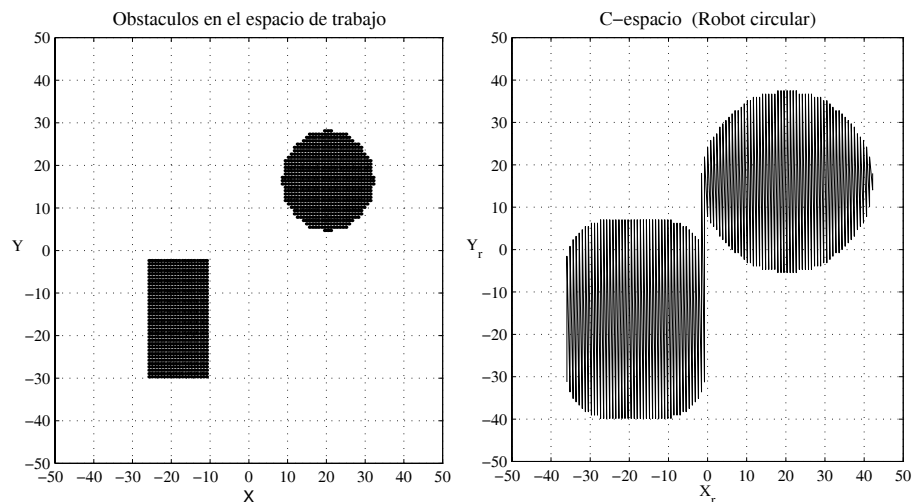


Figura 7.1: Espacio de trabajo y de las configuraciones para un robot móvil circular

- Robot Móvil Circular. Se presentan sus resultados por dos razones básicas. En primer lugar se trata de un robot en el que la idea sobre la que se sustenta este trabajo aparece de forma más intuitiva. Sirve para mostrar por otra parte el escaso tiempo de computación necesario para el cómputo de la representación de los obstáculos en el espacio de las configuraciones. Se ha considerado un espacio de trabajo de $1m \times 1m$ y un robot con forma circular de $10cm$ de radio. En el espacio de trabajo se han considerado dos obstáculos de formas rectangular y circular (figura ??). El espacio de las configuraciones obtenido se muestra en la figura ?. Como se puede observar los C-obstáculos se corresponde con un *crecimiento* de los obstáculos con el radio del robot en todas las direcciones. Este crecimiento está producido por la convolución de las dos funciones.

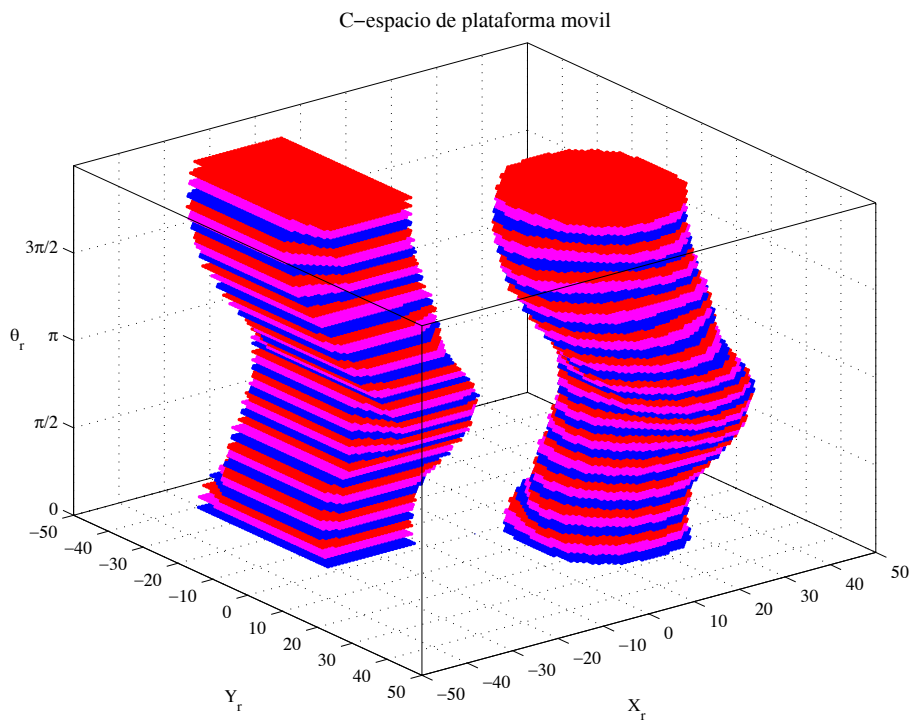


Figura 7.2: Espacio de las configuraciones para un robot móvil rectangular

- Plataforma rectangular. Las dimensiones elegidas para el robot rectangular son $5cm \times 10cm$. Se han considerado los mismos obstáculos que en el caso previo. El espacio de las configuraciones que se obtiene aparece re-

presentado en la figura ???. El resultado es semejante a otros que aparecen en la bibliografía, como es el que se ha descrito en la figura ??? de la página ???

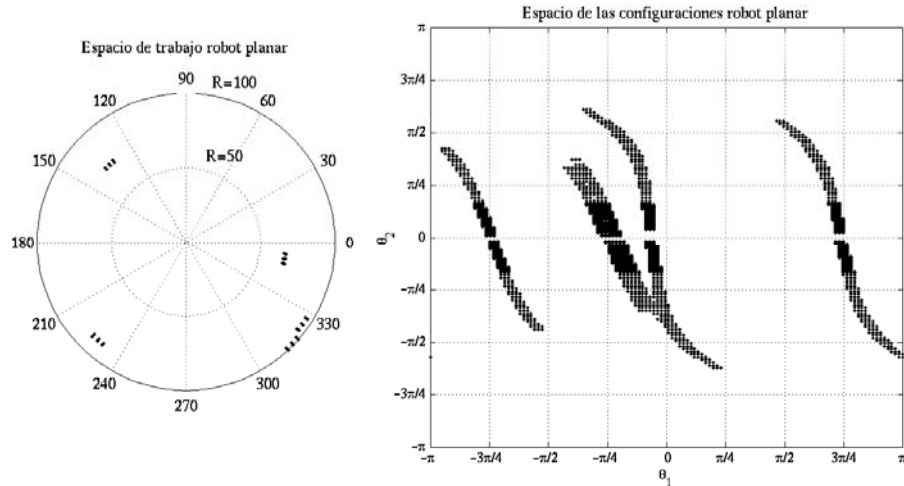


Figura 7.3: Espacio de trabajo y de las configuraciones para un robot planar

- Robot planar. Como se ha comentado en los capítulos anteriores, la idea de la convolución de dos mapas de bits donde se representa al robot y a los obstáculos, respectivamente, sólo era aplicada en la bibliografía para robots móviles. Para este tipo de robots la idea resulta bastante intuitiva. Sin embargo, para articulados no existen trabajos en esta línea. En secciones previas se han descrito los algoritmos que permiten computar el conjunto de configuraciones que producen colisión del robot con los obstáculos del espacio de trabajo. Para este caso concreto, se ha elegido un robot que consta de dos elementos formados por segmentos de línea con longitudes de 60cm y 50cm , respectivamente. Se han considerado cinco obstáculos puntuales distribuidos de forma aleatoria por el espacio de trabajo. Para cada uno de ellos se obtiene la típica curva con forma de S invertida (figura ???). Las dos configuraciones posibles *codo arriba* y *codo abajo* se distinguen por el ángulo θ_2 . Con esto se demuestra, por tanto, la validez del formalismo para cualquier tipo de robot y, en concreto, para el robot planar.

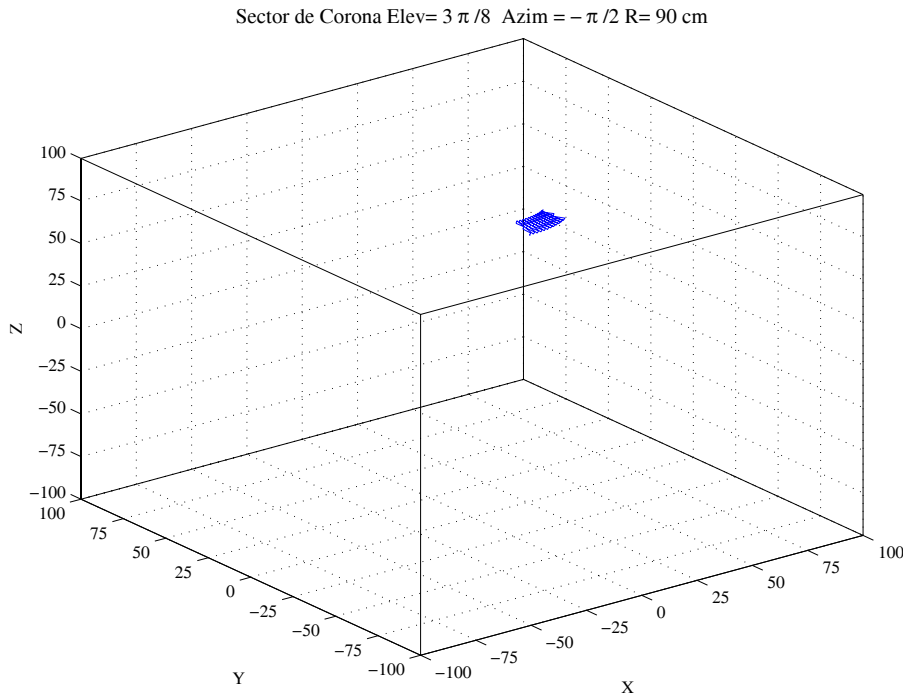


Figura 7.4: *Obstáculo 1* con forma de sector circular

- Robot PUMA. Este tipo de robot constituye un elemento de prueba válido y de gran interés para mostrar el carácter general del formalismo propuesto. El robot considerado está formado por tres elementos cuyas longitudes son 70cm , 60cm y 55cm , respectivamente. Se han elegido dos espacios de trabajo diferentes en su estudio. En primer lugar (*Obstáculo 1* figura ??), se ha considerado un sector circular situado a 90cm del hombro. El origen de coordenadas se ha considerado en la articulación del hombro. Se ha definido de una anchura (ángulo de azimut) pequeña para poder obtener curvas semejantes a las que se obtienen en el robot planar para un determinado valor de la cintura. A continuación se ha diseñado un obstáculo (*Obstáculo 2*) en forma de barrera paralela al eje X situada a 90cm con dimensiones $60\text{cm} \times 10\text{cm} \times 50\text{cm}$. Este obstáculo tiene un carácter más realista y aparece representado en la figura ??

Resulta útil representar los valores de las coordenadas esféricas de los puntos que constituyen el obstáculo con forma de barrera (figura ??). Su

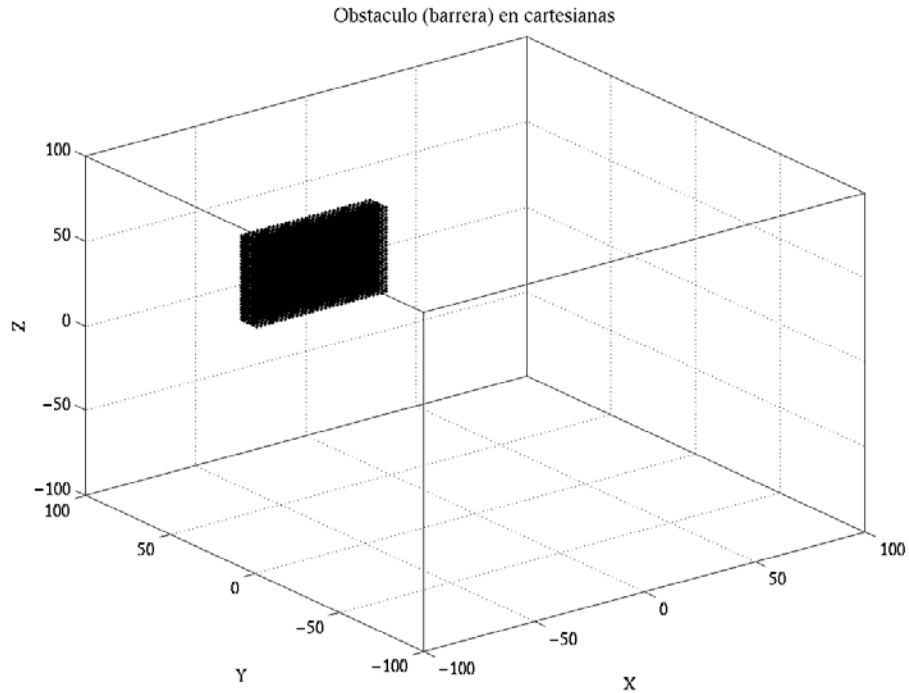


Figura 7.5: *Obstáculo 2* con forma de barrera

interés se debe a que se utilizan estas coordenadas para realizar el producto de convolución de las funciones que representan al robot y a los obstáculos. En la figura ?? se tiene la representación del C-obstáculo que genera la porción de corona. En ella se pueden observar dos siluetas con forma de S invertida con lo que queda mostrada la posibilidad de obtener las cuatro configuraciones posibles del robot para que exista un contacto con el obstáculo: *codo arriba/abajo*, *configuración directa/inversa*. Para cada una de estas últimas se asocia una de las dos curvas y la posición de *codo arriba/abajo* se diferencia analizando el valor del ángulo θ_3 de la tercera articulación. En la figura ?? aparecen los dos conjuntos de configuraciones que producen colisión con el robot para el *Obstáculo 2*. La obtención de estas representaciones puede ser considerada como una de las principales aportaciones de este trabajo de tesis doctoral.

En la tabla ?? se muestran los tiempos de cálculo obtenidos en la ejecución de los programas para la plataforma seleccionada. Los tiempos son bastante

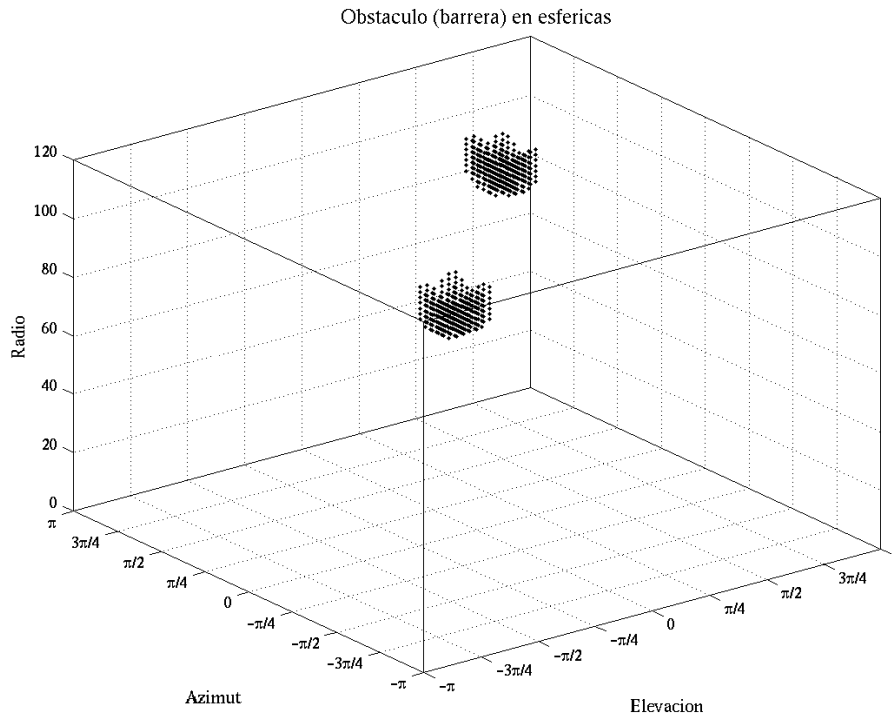


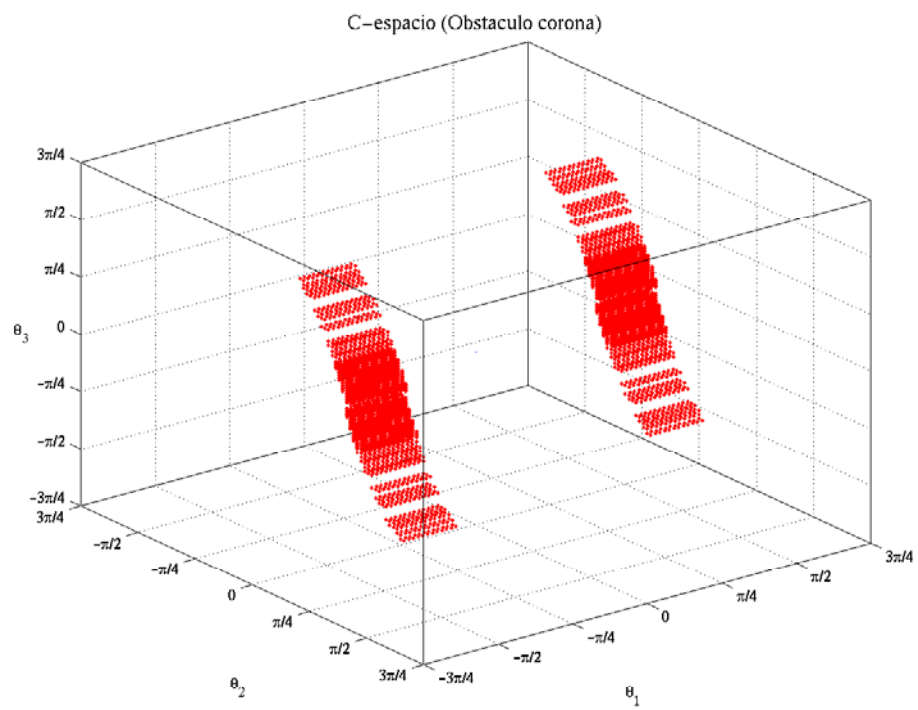
Figura 7.6: Coordenadas de los puntos del *Obstáculo 2* en esféricas

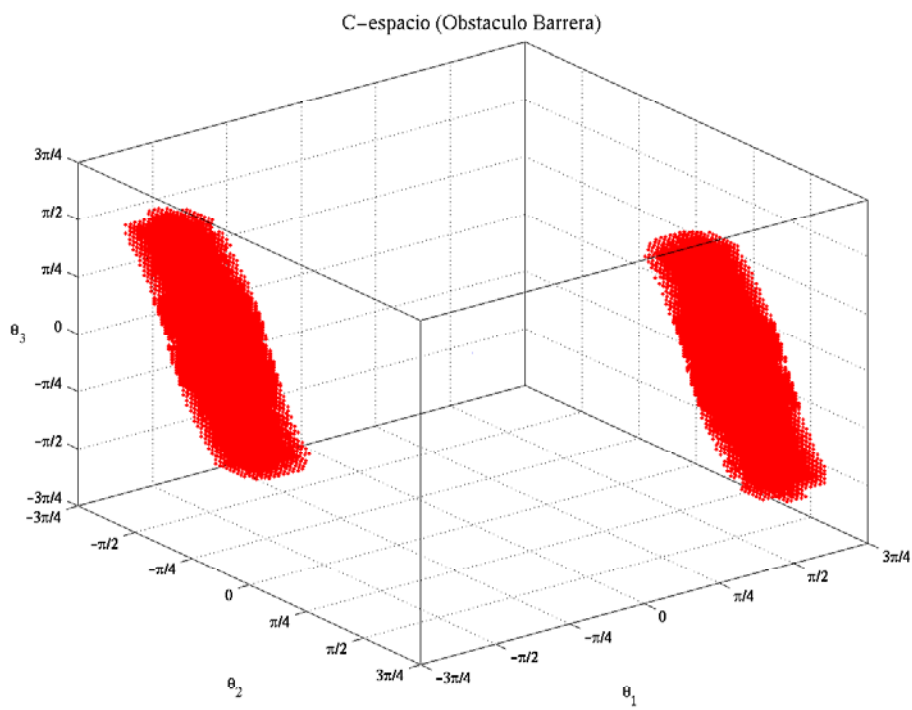
Cuadro 7.1: Tiempos de cálculo de los algoritmos

	Robot Circular	Robot Móvil	Robot Planar	Robot PUMA ¹
N=64	–	–	–	19.97 s/23.17 s
N=128	0.09 s	7.36 s	3.04 s	324.78 s/375.04s

reducidos aunque, como ya se ha mencionado, en este trabajo de investigación no se ha hecho especial hincapié en su disminución. La diferencia de tiempos para los dos obstáculos en el espacio de trabajo del PUMA puede ser debida al coste computacional asociado a la creación de las matrices *sparse* y a la naturaleza optimizada de la librería proporcionada por el fabricante que ha sido utilizada. Como trabajo futuro queda la implementación de los algoritmos propuestos sobre plataformas de cálculo de alto rendimiento.

¹Tiempos de cálculo para *Obstáculo 1/Obstáculo 2*

Figura 7.7: C-obstáculo para el *Obstáculo 1*

Figura 7.8: C-obstáculo para el *Obstáculo 2*

Capítulo 8

Cálculo de Mapa de Potenciales

El método que se ha propuesto en los capítulos previos para la proyección de los obstáculos al espacio de las configuraciones de un robot tiene como punto clave la expresión CB

$$CB(q) = \int A(q, x)B(x)dx \quad (8.1)$$

Se trata de una expresión integral, donde aparece el producto de una función que define al robot $A(q, x)$ y otra a los obstáculos $B(x)$ en el espacio de trabajo. En este capítulo se va a intentar encontrar una interpretación física para esta expresión, utilizando como herramienta matemática la teoría de las funciones de Green.

Estas funciones permiten la transformación de una ecuación diferencial en una ecuación integral. Bajo este planteamiento matemático, se puede pensar que $CB(q)$ es la solución de una ecuación diferencial, siendo $A(q, x)$ la función de Green asociada al operador diferencial.

Como se mostrará, desde un punto de vista físico, se puede interpretar que dicha ecuación diferencial relaciona una función de potencial artificial con la fuente que genera ese potencial. Hay que tener en cuenta que la técnica de los campos de potencial artificial ha sido ampliamente utilizada en Robótica.

En la siguiente sección de este capítulo se describirán dos tareas implicadas

en la consecución de robots autónomos, la planificación de caminos y el control reactivo, donde se ha aplicado el esquema de los campos de potencial artificial. A continuación, se revisarán los principales trabajos que han contribuido a desarrollar el planteamiento de los campos de potencial tanto en planificación local como global. Posteriormente, se presentan los conceptos fundamentales de la teoría de las funciones de Green. Esta técnica matemática proporcionará la base para obtener una interpretación de los C-obstáculos desde el punto de vista de los campos de potencial artificial. Este será el objetivo de las restantes secciones del capítulo. Para comprobar la validez de esta interpretación, además, se ha aplicado a un robot particular, considerando dos distribuciones de potencial artificial.

8.1. Control Reactivo y Planificación de Caminos

Una de las características de un sistema robótico inteligente, como ya se ha comentado, es la capacidad moverse a una posición deseada sin que se produzcan choques con los objetos que se encuentran en su entorno de trabajo. En general, existen dos planteamientos para abordar el problema, el control reactivo o planificación basada en sensores y la planificación de caminos o basada en modelos. En [?] se puede encontrar un análisis detallado de ambos planteamientos.

El primero engloba a un conjunto de técnicas cuya finalidad es determinar el movimiento deseado del robot en tiempo real examinando el modelo actual del entorno. Según el modelo varía el robot reacciona. Normalmente, se utilizan sensores para disponer del modelo actual. Además, el modelo tendría un carácter local, ya que depende de la información actual de los sensores y no contiene el estado global. Este planteamiento se aplica para evitar colisiones en tiempo real en entornos inciertos y cambiantes.

La planificación de caminos, sin embargo, trata de resolver el siguiente problema: dado un modelo perfecto del robot en un entorno estático y dos configuraciones, una inicial y otra objetivo, encontrar un camino libre de colisiones entre ambas configuraciones. Esta secuencia de movimientos se determina antes de que el robot comience a moverse. Cualquier robot que puede ejecutar de forma autónoma tareas de movimiento tendrá que resolver en algún grado

este problema. Sin embargo, el coste computacional asociado con este método es elevado. Por eso, ha habido un gran interés en desarrollar planificadores rápidos donde la idea del espacio de las configuraciones ha jugado un papel decisivo.

Estos dos planteamientos no tienen que ser excluyentes a la hora de abordar la construcción de un sistema robótico inteligente. Resultaría ideal poder combinar el control reactivo y la planificación de caminos, aprovechando las ventajas que ambos presentan. Por un lado, la planificación de caminos proporciona la capacidad de mover al robot para lograr objetivos complejos. Si en su entorno hay muchos obstáculos, cada paso de la secuencia de movimientos deberá realizarse correctamente. Por otro, el control reactivo proporciona un rendimiento robusto para tratar con obstáculos no esperados e inciertos mientras se ejecuta el movimiento planificado. Sin embargo, el problema de combinar ambos métodos es que el camino planificado especifica exactamente por dónde debería moverse el robot para alcanzar el objetivo, y un controlador reactivo necesitará mover al robot fuera del camino original. No obstante, se han propuesto distintas técnicas para combinar ambos planteamientos. Entre ellas, se encuentra los campos de potencial artificial que se utiliza, además de para aunar las dos vertientes, en cada una de ellas por separado.

8.2. Campos de Potencial Artificial

Una de las técnicas que se encuentra dentro de este esquema es la de los campos de potencial artificial. La introducción de este concepto se debe a Khatib [?] que, inicialmente, propuso su implementación para evitar los obstáculos en tiempo real. La idea fundamental de este planteamiento es considerar que el robot se está moviendo dentro de un campo de fuerzas. Estas fuerzas están formadas por dos componentes: una de atracción que empuja el robot hacia la configuración final y otra repulsiva que lo aparta de los obstáculos dentro de su espacio de trabajo. Si estas fuerzas se aplican a los actuadores del robot, éste se mueve hacia la configuración destino sin chocar con los obstáculos. No se calcula de forma específica ningún camino.

El método de los campos de potencial sirve también para mejorar el control de bajo nivel. En vez de que el sistema de control únicamente siga una trayectoria

dada, se tienen en cuenta los obstáculos y el robot responde de acuerdo con ello. Como esta estrategia para evitar obstáculos es sencilla, se puede implementar de forma eficiente e incorporarla directamente al lazo de tiempo real del sistema de control.

En esta técnica, la fuerza a la que se encuentra sometido el robot se especifica como

$$F(q) = -\nabla V(q)$$

donde $V(q)$ es una función escalar no negativa definida sobre el espacio de las configuraciones del robot. Se puede considerar que la función V especifica la energía potencial del robot en una configuración y de aquí surge el nombre con que se denomina a esta técnica.

Una ventaja que se obtiene al especificar la fuerza como el gradiente de un potencial es que se puede predecir el comportamiento del sistema sin conocer la secuencia exacta de configuraciones que seguirá el robot. En concreto, si el espacio de las configuraciones está limitado y la función potencial es invariante en el tiempo, se puede demostrar [?] que el robot se moverá hacia un mínimo local del potencial.

Un requisito clave del análisis previo es que la función potencial sea invariante en el tiempo. Sin embargo, en las aplicaciones típicas de este método, el potencial es una función del entorno. De esta forma la invarianza en el tiempo se corresponde con entornos estáticos. Pero incluso en el caso donde no se cumpla esta condición, este planteamiento para definir fuerzas todavía intenta que el robot se esté moviendo continuamente en una dirección adecuada.

Otra propiedad importante de las funciones de potencial es su carácter aditivo. Se pueden considerar independientemente diferentes potenciales para dotar al robot de distintas capacidades. Si se utiliza la suma de dichos potenciales para controlar el robot, su comportamiento tratará de combinar las capacidades deseadas. Por ejemplo, un potencial intentaría mover al robot hacia la configuración destino, mientras que otro evitaría la colisión con los objetos. Si se combinan estos potenciales se logra un comportamiento de búsqueda del objetivo y evitar choques. Si posteriormente se desea evitar los límites internos de las articulaciones, entonces se puede incluir esta capacidad diseñando un potencial adecuado y sumarlo a la combinación previa.

Sin embargo, si se suman potenciales cuyas finalidades puedan interferir entre ellas, el robot podría quedar atrapado en un mínimo local. En este caso podría haber configuraciones en que la suma de las fuerzas asociadas con los potenciales sea nula y como consecuencia el robot no se mueva. Los mínimos locales impiden que el método del campo de potenciales resuelva directamente el problema del movimiento libre de colisiones de los robots.

Dado que este método permite guiar al robot hacia la configuración deseada con un coste computacional mínimo asociado a la representación del espacio libre, surgió un enorme interés en adaptarlo a la planificación de caminos. La mayor dificultad que aparece es que según se sigue el sentido opuesto al gradiente del potencial se llegue a un punto en el que se alcance un mínimo local del potencial, distinto del punto objetivo, del que fuera difícil escapar. Este problema se puede resolver si se utilizan funciones de potencial especiales, denominadas *funciones de navegación*, que no presentan mínimos locales salvo el final. Se han propuesto funciones de navegación analíticas [?], aunque para entornos muy concretos. También se han desarrollado funciones de navegación numéricas [?] definidas sobre una representación discreta del espacio de las configuraciones. Trabajando sobre un espacio de las configuraciones discreto se ha propuesto en [?] [?] utilizar movimientos aleatorios para escapar de los mínimos locales, que se alcanzan siguiendo el gradiente negado del potencial. En otros trabajos se propone utilizar de forma conjunta los campos de potencial con otras técnicas de planificación, entre las que se encuentra el mapa de carreteras. En [?], por medio de un mapa de carreteras, se calcula el campo de potencial sin mínimos locales en el espacio de trabajo. A partir de él se obtiene el campo de potencial en el espacio de las configuraciones, aunque ya no se garantiza que no existan mínimos locales y, por tanto, se utilizan herramientas potentes para salir de ellos. El planteamiento propuesto en [?] es más directo porque, a partir de los C-obstáculos, se calcula el campo de potencial en el espacio de las configuraciones. De forma simultánea se obtiene el mapa de carreteras en el C-espacio, que permite simplificar la búsqueda. En la etapa de planificación se utilizan algoritmos de exploración de grafos para guiar la búsqueda. En la definición de su heurística se incluye el potencial con el fin de evitar las zonas angostas. En [?] se utilizan las funciones armónicas para garantizar que no existen mínimos

locales y que el objetivo es alcanzable.

8.3. Funciones de Green

El objetivo fundamental de este capítulo es encontrar una nueva interpretación para la expresión $??$. En ella aparece una integral extendida sobre el espacio de trabajo W . Existe una relación estrecha entre los operadores diferenciales e integrales, donde las condiciones límite juegan un papel fundamental. La conexión entre ambos es esencialmente la teoría de las funciones de Green. En esta sección no se pretende tratar de forma profunda este tema, aunque se intentará mostrar las ideas principales. En [?] se encuentra un tratamiento matemático riguroso y su utilización en diferentes campos de la Física.

8.3.1. Introducción

La ecuación diferencial más sencilla es

$$-\frac{dy}{dx} = f(x) \quad (8.2)$$

Si se elige como condición inicial $y(a) = y_0$, su solución se puede escribir como

$$y(x) = y_0 + \int_a^x f(x') dx' \quad (8.3)$$

De esta forma se resuelve esta sencilla ecuación diferencial utilizando la integración. Se puede complicar el problema un poco más, suponiendo que x se encuentra en el intervalo cerrado $[a, b]$ ($x \in [a, b]$). Entonces la ecuación $(?)$ se puede escribir como

$$y(x) = y_0 + \int_a^b \vartheta(x - x') f(x') dx' \quad (8.4)$$

donde $\vartheta(x)$ es la función salto definida como

$$\vartheta(x) = \begin{cases} 1 & \text{si } x > 0 \\ 0 & \text{si } x \leq 0 \end{cases} \quad (8.5)$$

Al introducir la función salto en el integrando de la ecuación se corta la integración sobre x' en $x = x'$. Así la ecuación $(?)$ es equivalente a la ecuación $(?)$. Cabe resaltar que $y(x)$ se puede poner como

$$y(x) = y_0 + K f(x) \quad (8.6)$$

donde K es un operador integral definido como

$$Kf(x) \equiv \int_a^b \vartheta(x-x')f(x')dx' \quad (8.7)$$

Aquí $\vartheta(x-x')$ es el *kernel* del operador integral K , y cuando el kernel proviene de la solución de una ecuación con un operador diferencial se le denomina normalmente como la *función de Green* de ese operador diferencial *para las condiciones límite apropiadas*. Así

$$G_1(x, x') = \vartheta(x-x') \quad (8.8)$$

es la función de Green perteneciente a d/dx para un sistema sujeto a la condición límite $y(a) = y_0$.

El método anterior puede resultar un poco rebuscado para resolver una ecuación diferencial tan simple. Con objeto de mostrar la simplificación que supone el utilizar las funciones de Green, se van a aplicar para resolver una ecuación diferencial que no es integrable de forma inmediata. Sea la ecuación

$$-\frac{dy}{dx} = f(x, y) \quad (8.9)$$

Entonces su solución sería

$$y(x) = \alpha + \int_0^1 G(x, x')f[x', y(x')]dx' \quad (8.10)$$

Como la función $y(x)$ aparece ahora bajo el signo integral, entonces se ha transformado la ecuación diferencial en una ecuación integral. Las funciones de Green permiten la transformación de una ecuación diferencial en una ecuación integral. Con ello es posible realizar simplificaciones matemáticas significativas para un problema determinado.

8.3.2. Las Funciones de Green y las Funciones Delta

A continuación se va a mostrar una forma interesante de ver la función de Green, que está relacionada con la función delta de Dirac $\delta(x-x')$. Sea la ecuación

$$Ly = f \quad (8.11)$$

donde L es un operador diferencial lineal y f es una función dada. En términos de la función de Green, su solución sería

$$y(x) = \int G(x, x')f(x')dx' \quad (8.12)$$

donde $G(x, x')$ es la función de Green asociada con el operador diferencial L . Si se aplica L sobre $G(x, x')$ se obtiene que

$$LG(x, x') = \delta(x - x') \quad (8.13)$$

También se puede poner como

$$LK = I \quad (8.14)$$

donde I cumple la propiedad de la δ

$$\int I(x, x')f(x')dx = f(x) \quad (8.15)$$

y K es el operador integral definido por

$$Kf = \int G(x, x')f(x')dx' \quad (8.16)$$

A partir de la ecuación (??), se puede observar que encontrar la función de Green de un operador L es equivalente a invertir L . De esta forma las ecuaciones del tipo $Ly = f$ se pueden resolver como

$$y = L^{-1}f = Kf = \int G(x, x')f(x')dx' \quad (8.17)$$

Este análisis se puede comprobar con la función de Green más sencilla $G_1(x, x')$ asociada al operador d/dx . Si se aplica el operador a la función

$$-\frac{d}{dx}G_1(x, x') = \delta(x - x') \quad (8.18)$$

Utilizando la ecuación (??) se obtiene

$$\frac{d}{dx}\theta(x - x') = \delta(x - x') \quad (8.19)$$

En general, las funciones de Green satisfacen

$$LG(x, x') = \delta(x - x') \quad (8.20)$$

A la vista de esta característica de las funciones de Green, si se puede obtener una función G que satisfaga

$$LG(x, x') = \delta(x - x') \quad (8.21)$$

entonces, si también se tienen las soluciones ε de la ecuación homogénea $L\phi = 0$, la solución general de

$$L\phi = F(x) \quad (8.22)$$

se puede escribir como

$$\phi(x) = \varepsilon(x) + \int G(x, x')F(x')dx' \quad (8.23)$$

Una aplicación interesante de las funciones de Green se presenta en la teoría electromagnética. La ecuación de Poisson permite calcular el potencial electromagnético espacial $\phi(r)$ generado por una distribución de carga $\rho(r)$:

$$\nabla^2\phi(r) = -4\pi\rho(r) \quad (8.24)$$

Utilizando el método de la transformada de Fourier se resuelve la ecuación

$$\nabla^2G(r, r') = \delta(r - r') \quad (8.25)$$

y se llega a que la función de Green correspondiente es

$$G(r, r') = -\frac{1}{4\pi} \frac{1}{|r - r'|} \quad (8.26)$$

En este caso la función de Green representa el potencial generado por una carga puntual situada en r' . Con lo expuesto anteriormente (expresión ??) se tiene

$$\phi(r) = \int \frac{\rho(r')}{|r - r'|} d^3r' \quad (8.27)$$

Haciendo un desarrollo en serie y considerando $|r|$ suficientemente grande se llega a la conocida expresión

$$\phi(r) \rightarrow \frac{Q}{r}$$

donde

$$Q = \int \rho(r')d^3r'$$

8.4. Interpretación de los C-obstáculos con las Funciones de Green

La teoría de las funciones de Green se puede aplicar para interpretar, desde un punto de vista diferente, la expresión propuesta en esta tesis para calcular la proyección de los obstáculos en el espacio de las configuraciones.

$$CB(q) = \int A(q, x)B(x)dx \quad (8.28)$$

Comparando esta expresión con ??, que es la solución de la ecuación diferencial ??, debería existir un operador diferencial L tal que

$$L[CB(q)] = B(x) \quad (8.29)$$

y cuya función de Green G asociada fuera $A(q, x)$, que debe cumplir

$$L[A(q, x)] = \delta(q, x) \quad (8.30)$$

No se trata de encontrar el operador diferencial L que satisfaga ??, ya que éste dependerá de cuál sea la función $A(q, x)$ que, como cabe recordar, definía al robot en una determinada configuración y situado en un determinado punto de su espacio de trabajo. Se trata de relacionar la expresión integral con una ecuación diferencial para poder buscarle una interpretación desde un punto de vista físico. Como es bien conocido, las ecuaciones diferenciales se utilizan ampliamente en los diferentes campos de la Física para modelar el comportamiento de los sistemas.

8.5. Interpretación de los C-obstáculos con los Campos de Potencial Artificial

Una vez que se ha establecido la base matemática, se está ya en condiciones de buscarle una analogía con alguna de las magnitudes físicas.

Tomando como punto de partida la ecuación diferencial ??, se podría pensar en la semejanza que existe con la ecuación de Poisson ($\nabla^2\phi(r) = -4\pi\rho(r)$), donde se define la distribución de potencial eléctrico ϕ originada por una distribución de carga eléctrica ρ .

De forma general se puede interpretar que CB se corresponde con una distribución de potencial artificial y que el obstáculo $B(x)$ es la causa que origina esa distribución. Asimismo, con la expresión ??, el robot $A(q, x)$ sería el potencial artificial asociado a una fuente de potencial puntual, que en este caso correspondería con un obstáculo puntual.

En cada ecuación diferencial aparece un operador diferencial L concreto, asociado con una función de Green, que establece la forma de la solución de la ecuación diferencial. Así, en la ecuación de Poisson el operador diferencial L es

∇ , siendo su función de Green

$$G(r, r') = -\frac{1}{4\pi} \frac{1}{|r - r'|}$$

que conduce a que la solución de la ecuación sea

$$\phi(r) = \int \frac{\rho(r')}{|r - r'|} d^3 r'$$

No obstante, en algunas ocasiones resulta más interesante establecer la función de Green asociada al operador diferencial L que fijar dicho operador. Ambos planteamientos conducirían a la misma solución. Por tanto, definiendo la forma de dicha función de Green $A(q, x)$ se obtendría la distribución de potencial, asociada a los obstáculos, con el comportamiento deseado. Esta manera de proceder es la que se ha elegido en las siguientes secciones.

8.6. Aplicación a un Robot Rígido Móvil

Para comprobar la validez de la interpretación planteada se va a aplicar a un robot particular. En concreto, se va a considerar que el robot es un objeto rígido con forma circular que se mueve libremente en R^2 . Así, se tiene que $q = (x_r, y_r)$ y $A(q, x) = A(x_r, y_r, x, y)$. Con ello, la expresión para el cálculo de los C-obstáculos venía dada por

$$CB(x_r, y_r) = \int A(x_r, y_r, x, y) B(x, y) dx dy \quad (8.31)$$

Con la teoría de las funciones de Green como base, debe existir un operador diferencial L tal que

$$L[CB(x_r, y_r, x, y)] = B(x, y)$$

siendo $A(x_r, y_r, x, y)$ su función de Green asociada, que debe cumplir que

$$L[A(x_r, y_r, x, y)] = \delta(x_r - x, y_r - y)$$

De esta manera, $CB(x_r, y_r, x, y)$ es el potencial artificial generado por una distribución de fuentes $B(x, y)$, que se corresponden con los obstáculos. La forma concreta de la distribución del potencial artificial depende de la elección de $A(x_r, y_r, x, y)$, que sería el potencial debido a una fuente puntual situada en (x_r, y_r) . A continuación, se van a proponer dos definiciones para la función de Green $A(q, x)$.

En primer lugar se considerará un potencial barrera, de forma que $A(q, x)$ vendría definida por

$$A(x_r, y_r, x, y) = \begin{cases} V & \text{si } (x, y) \in \mathbf{A}_{(x_r, y_r)} \\ 0 & \text{si } (x, y) \notin \mathbf{A}_{(x_r, y_r)} \end{cases} \quad (8.32)$$

donde $\mathbf{A}_{(x_r, y_r)}$ es el subconjunto de puntos del espacio de trabajo que representa al robot en la configuración (x_r, y_r) , y V es una constante. Si $V = 1$ la situación planteada coincide con la proyección de los obstáculos en el espacio de las configuraciones de un robot circular. Esto se puede comprobar en la figura ?? que ya ha sido analizada en el capítulo anterior.

Observando esta figura se puede interpretar que la proyección de los obstáculos en el C-espacio puede ser vista como una distribución de potencial artificial con la hipótesis de que un obstáculo puntual genera un potencial de barrera. Concretamente, los efectos de este potencial aparecen en los límites de la región de los C-obstáculos y son nulos en el resto. La utilidad de este potencial sería para la detección de colisiones.

Otra de las posibles definiciones del potencial es aquella que, de forma similar con el electrostático, decrece a medida que se aleja del límite de los obstáculos. Así, la función que genera el potencial vendría dada por

$$A(x_r, y_r, x, y) = \begin{cases} \infty & \text{si } (x, y) \in \mathbf{A}_{(x_r, y_r)} \\ V/r & \text{si } (x, y) \notin \mathbf{A}_{(x_r, y_r)} \end{cases} \quad (8.33)$$

donde V es una constante y r es la distancia mínima del punto al robot. Si en el algoritmo propuesto para el cálculo de los C-obstáculos de un robot circular se utiliza una matriz bidimensional que siga esta distribución en lugar de utilizar la típica matriz binaria para definir el robot, se obtendría el campo de potencial artificial generado por los obstáculos para esa distribución concreta.

Así, para un robot circular con un radio de 10cm cuyo espacio de trabajo de $1\text{m} \times 1\text{m}$ se encuentra ocupado por dos obstáculos, uno rectangular y otro circular, se puede definir su función de Green $A(x_r, y_r)$ (?). En la figura ?? se pueden observar el espacio de trabajo y la función de Green considerada. Se puede interpretar como ésta última se puede asociar al potencial que se generaría considerando un obstáculo puntual en el origen.

Procediendo a la aplicación del algoritmo se obtiene directamente la superficie de potencial generada por los obstáculos de este espacio de trabajo. El

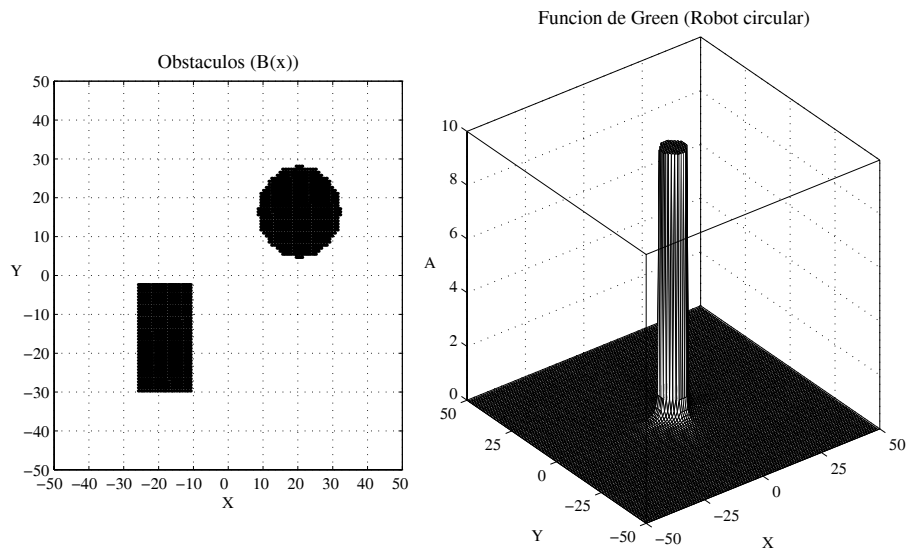


Figura 8.1: Espacio de trabajo y función de Green considerada

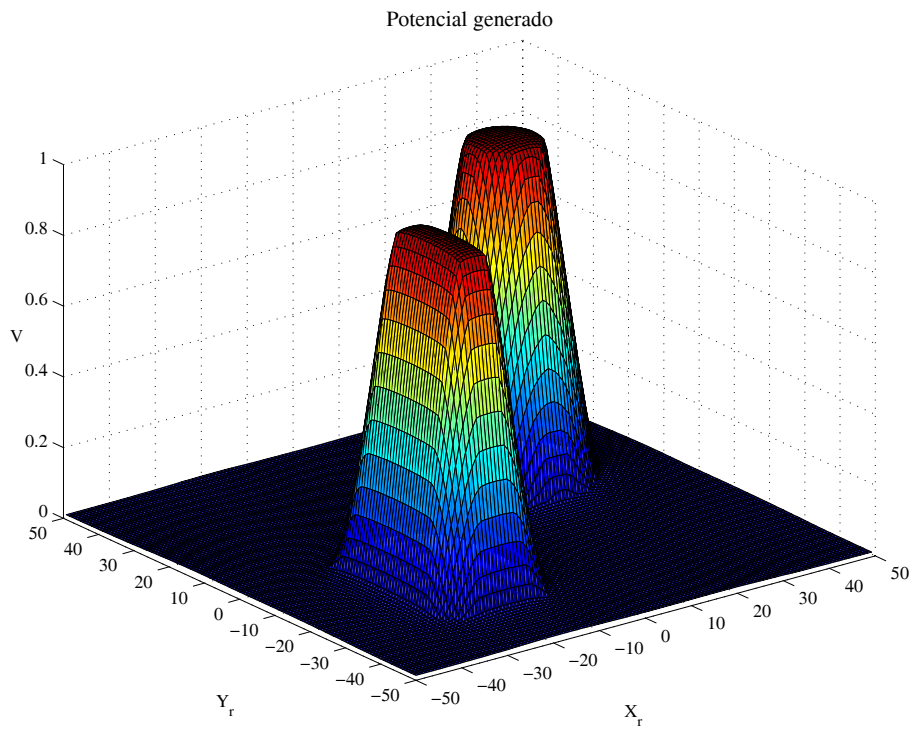


Figura 8.2: Superficie de potencial en el C-espacio

resultado aparece en la figura ???. En ella, se puede observar como los obstáculos generan un potencial repulsivo que decrece con la distancia, a semejanza con lo que ocurre con el potencial eléctrico. Esta figura muestra la validez de la interpretación física propuesta así como su utilidad para los procedimientos de planificación que utilizan campos de potencial artificial.

Los resultados que se obtienen coinciden con los que aparecen en los distintos trabajos de investigación que se ocupan de este tema. La gran ventaja que ofrece este nuevo planteamiento es que, utilizando la misma expresión que en el cálculo de los C-obstáculos, directamente se puede obtener el potencial en el espacio de las configuraciones sin necesidad de etapas intermedias, como se realiza en otros trabajos. Esto supone una reducción del tiempo de cálculo asociado con los algoritmos y es uno de los principales resultados de este trabajo de tesis.

Capítulo 9

Conclusiones y Trabajos

Futuros

Las principales aportaciones y conclusiones de este trabajo de tesis son las siguientes:

- Se ha propuesto un nuevo formalismo matemático para el cálculo de los obstáculos en el espacio de las configuraciones de un robot. La expresión central de este formalismo permite calcular los C-obstáculos como la integral extendida sobre el espacio de trabajo del producto de dos funciones; una que describe al robot y la otra a los obstáculos.
- Se ha demostrado que la definición de C-obstáculo propuesta en este formalismo es equivalente a otras expresiones de la bibliografía. Esta nueva definición de C-obstáculo presenta la ventaja de tratar con objetos generales y reales, sin necesidad de realizar ninguna aproximación.
- Basado en el formalismo matemático se ha presentado un método de trabajo que tiene un carácter general y que es aplicable a cualquier estructura robótica. Con la elección adecuada de los sistema de referencia y las coordenadas de trabajo, directamente relacionados ambos con la estructura mecánica del robot, el método deriva en que el C-espacio es la convolución de la función que representa al robot y la que representa a los obstáculos espacio de trabajo

- El método se ha validado sobre un conjunto de problemas variado, cubriendo los campos de la Robótica móvil y los manipuladores. Así, en primer lugar, se ha aplicado a un objeto rígido bidimensional que se mueve sobre un plano, llegando a las mismas expresiones que las que aparecen en la bibliografía. En segundo lugar, se ha extendido a un robot móvil tridimensional que se desplaza sobre un plano, obteniéndose los mismos resultados que aparecen en otros trabajos de investigación.
- Posteriormente, se han considerado robots articulados en espacios de trabajo bidimensionales y tridimensionales. Se ha obtenido una representación explícita de los C-obstáculos para los manipuladores más ampliamente utilizados en la industria, como son el PUMA, el SCARA y el Stanford. Esta ha sido una de las mayores aportaciones de este trabajo de tesis, ya que existen pocos trabajos que calculen los C-obstáculos para manipuladores industriales.
- Se ha puesto de manifiesto que la complejidad computacional del método en todos los casos es independiente de la forma de los objetos que constituyen el robot o de los que están presentes en su entorno de trabajo. Con ello se ha logrado, por un lado, que la complejidad computacional no dependa del número de vértices de los objetos ni de si son cóncavos o convexos, como sucede en la mayoría de los trabajos existentes. Por otro lado, se ha conseguido obtener una representación exacta de los C-obstáculos, ya que no ha sido necesario realizar ninguna aproximación en cuanto a la forma de los objetos.
- La propuesta teórica se ha completado con la consideración de aspectos prácticos de implementación para reducir la complejidad computacional. Así, se ha utilizado como herramienta matemática la transformada de Fourier en lugar de la convolución y, como solución algorítmica la Transformada Rápida de Fourier.
- Se han propuesto y se han implementado los algoritmos para la obtención de los C-obstáculos tanto de robots móviles como de manipuladores articulados. Los algoritmos propuestos trabajan con mapas de bits que representan al robot y a los obstáculos en su espacio de trabajo, y obtienen

un mapa de bits que representa al espacio de las configuraciones. El tiempo de ejecución de estos algoritmos depende, en cada caso, únicamente de la resolución con la que se trabaje y es independiente del número de obstáculos y de su forma geométrica.

- Debido al carácter paralelizable de los algoritmos que calculan la transformada rápida de Fourier y a la fácil paralelización de los algoritmos propuestos se reduce de forma drástica el tiempo de cálculo, pudiendo aplicarse a entornos de trabajo que varían dinámicamente.
- Se ha proporcionado una interpretación física del formalismo, utilizando como herramienta matemática la teoría de las funciones de Green. Así, se ha demostrado que la expresión central del formalismo permite obtener el potencial generado por diversas fuentes, que son los obstáculos. Además, la función que describe al robot se ha identificado con el potencial creado por una fuente puntual situada en el origen.
- Se ha comprobado la validez de esta interpretación considerando un robot circular. Con los algoritmos desarrollados se han obtenido superficies de potencial con el comportamiento deseado. Como gran ventaja de esta forma de trabajo se ha destacado que no existe un gasto computacional asociado a la obtención de los campos de potencial artificial en el espacio de las configuraciones. Este aspecto supone otra de las principales aportaciones de este trabajo de tesis.

A partir de las conclusiones se pueden destacar algunos puntos en los que es posible extender el trabajo presentado:

- Implementar los algoritmos sobre dispositivos de cálculo especializados de alto rendimiento.
- Integrar la metodología en los procedimientos de planificación y control de trayectorias.
- Estudiar la nueva técnica para la obtención de potenciales artificiales para diferentes estructuras robóticas como los manipuladores industriales.
- Proponer y analizar nuevas funciones de potencial.

Bibliografía

- [AB88] F. Avnaim and J. D. Boissonnat. Polygon placement under translation and rotation. Technical Report 889, INRIA, Sophia-Antipolis, France, 1988.
- [ABF88] F. Avnaim, J. D. Boissonnat, and B. Faverjon. A practical exact motion planning algorithm for polygonal objects amidst polygonal obstacles. In *Proc. of the IEEE Conference on Robotics and Automation*. IEEE, 1988.
- [BF70] F. W. Byron and R. W. Fuller. *Mathematics of Classical and Quantum Physics*. Addison Wesley Publishing Co., Reading, MA, 1970.
- [BK88] C. Bajaj and M. S. Kim. Generation of configuration-space obstacles: the case the moving sphere. *IEEE Journal of Robotics and Automation*, 4(1), 1988.
- [BL83] R. A. Brooks and T. LozanoPérez. A subdivision algorithm in configuration space for findpath with rotation. In *Proceedings of the 8th International Conference on Artificial Intelligence*. 799-806, 1983.
- [BL91] J. Barraquand and J. C. Latombe. Robot motion planning: a distributed representation approach. *Int. J. of Robotics Research*, 10(6), 1991.
- [Bla97] F. J. Blanco. Un nuevo enfoque en la planificación de caminos libres de colisiones para estructuras robóticas. Memoria de grado, Universidad de Salamanca, 1997.

- [BLL89] J. Barraquand, B. Langlois, and J. C. Latombe. Robot motion planning with many degrees of freedom and dynamic constraints. In *Preprints of the Fifth International Symposium of Robotics Research*, pages 74–83, 1989.
- [BN90] M. S. Branicky and W. S. Newman. Rapid computation of configuration space obstacles. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 1990.
- [BPBA97] A. Barrientos, L. F. Peñin, C. Balaguer, and R. Aracil. *Fundamentos de Robotica*. McGraw Hill, Madrid, 1997.
- [Bro89] R. C. Brost. Computing metric and topological properties of configuration-obstacles. In *Proceedings of the IEEE Conf. on Robotics and Automation*, pages 170–176, 1989.
- [Can88] J. F. Canny. *The complexity of robot motion planning*. MIT Press, Cambridge, 1988.
- [Cra89] J. J. Craig. *Introduction to Robotics mechanics and control*. Addison Wesley Publishing Co., Reading, MA, 1989.
- [DM90] J. R. Dooley and J. M. McCarthy. Parameterized descriptions of the joint space obstacles for a 5r closed chain robot. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1542–1547, 1990.
- [Don84] R. Donald. Motion planning with six degrees of freedom. Technical Report AI-TR-791, Artificial Intelligence Laboratory MIT, 1984.
- [DX89] B. R. Donald and P. Xavier. A provably good approximation algorithm for optimal-time trajectory planning. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 958–963, 1989.
- [Fav86] B. Faverjon. Object level programming of industrial robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1406–1412,, 1986.

- [FT87] B. Faverjon and P. Tournassoud. A local based approach for path planning of manipulators with a high number of degrees of freedom. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1152–1159, 1987.
- [FT89] B. Faverjon and P. Tournassoud. A practical approach to motion planning for manipulators with many degrees of freedom. In *Preprints of the Fifth International Symposium of Robotics Research*, pages 65–73, 1989.
- [GM89] Q. J. Ge and J. M. McCarthy. Equations for boundaries of joint obstacles for planar robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 418–423, 1989.
- [GM90] Q. J. Ge and J. M. McCarthy. An algebraic formulation of configuration-space obstacles for spatial robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1542–1547, 1990.
- [Gou84] L. Gouzènes. Strategies for solving collisions-free trajectories problems for mobile and manipulator robots. *International Journal of Robotics Research*, 3(4):51–65, 1984.
- [GRS83] L. Guibas, L. Ramshaw, and J. Stolfi. A kinetic framework for computational geometry. In *Proc. of the IEEE Symposium on Foundations of Computer Science*, pages 100–111, 1983.
- [GS86] L. Guibas and R. Seidel. Computing convolution by reciprocal search. In *Proceedings of the ACM Symposium on Computational Geometry*, pages 90–99, 1986.
- [Her86] M. Herman. Fast, three-dimensional, collision-free motion planning. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1056–1063, 1986.
- [HS91] L. S. Homem and A. S. Sanderson. Two criteria for the selection of assembly plans: maximizing the flexibility of sequencing the assembly tasks and minimizing the assembly time through parallel execution

- of assembly tasks. *IEEE Transactions on Robotics and Automation*, 7(5), 1991.
- [Hwa90] Y. K. Hwang. Boundary equations of configurations obstacles for manipulators. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 298–303, 1990.
- [JC89] P. Jacobs and J. Canny. Planning smooth paths for mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2–7, 1989.
- [Kav94] L. E. Kavraki. *Random networks in configuration space for fast path planning*. Ph. d. dissertation, Stanford University, 1994.
- [Kav95] L. E. Kavraki. Computation of configuration- space obstacles using the fast fourier transform. *IEEE Tr. on Robotics and Automation*, 11(3), 1995.
- [KCN89] R. D. Klafter, T. A. Chmielewski, and M.Ñegin. *Robotic Engineering an integrated approach*. Prentice Hall, New Jersey, 1989.
- [Kha86] O. Khatib. Real-time obstacles avoidance for manipulators and mobile robots. *International Journal of Robotics Research*, 5(1):90–98, 1986.
- [Kod87] D. E. Koditscheck. Exact robot navigation by means of potential functions:some topological considerations. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1–6, 1987.
- [Lat91a] J. C. Latombe. A fast path planner for a car-like indoor mobile robot. In *Proceedings of the 9th Nat. Conference on Artificial Intelligence*, pages 659–665, 1991.
- [Lat91b] J. C. Latombe. *Robot motion planning*. Kluwer Academic Publishers, Boston, MA, 1991.
- [LGF86] C. S. Lee, R. C. Gonzalez, and K. S. Fu. *Tutorial on robotics*. IEEE Computer Society, Los Angeles, CA, 1986.

- [LO91] T. LozanoPérez and P. O'Donnell. Parallel robot motion planning. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 1000–1007, 1991.
- [Loz81] T. LozanoPérez. Automatic planning of manipulator transfer movements. *IEEE Transactions on Systems, Man and Cybernetics*, 11(10):681–698, 1981.
- [Loz83] T. LozanoPérez. Spatial planning: a configuration space approach. *IEEE Transactions on Computers*, 32:108–120, 2 1983.
- [Loz87] T. LozanoPérez. A simple motion-planning algorithm for general robot manipulators. *IEEE Journal of Robotics and Automation*, 3(3):224–238, 1987.
- [LRDG90] J. Lengyel, M. Reichert, B. R. Donald, and D. P. Greenberg. Real-time robot motion planning using rasterizing computer graphics hardware. In *Proceedings of SIGGRAPH'90*, pages 327–335, 1990.
- [Lum87] V. Lumelsky. Effect of kinematics on motion planning for planar robot arms moving amidst unknown obstacles. *IEEE Transactions on Robotics and Automation*, 3(3):207–223, 1987.
- [MF93] A. A. Maciejewski and J. J. Fox. Path planning and the topology of configuration space. *IEEE Tr. on Robotics and Automation*, 9(4), 1993.
- [MJG97] I. Mantegh, M. Jenkin, and A. Goldenberg. Solving the find-path problem: a complete and less complex approach using the bie methodology. In *Proceedings of the Symposium an Computers Intelligent IEEE CIRA*, 1997.
- [Mor96] V. Moreno. *Planificación de trayectorias utilizando algoritmos de exploración de grafos en paralelo*. Ph. d. dissertation, Universidad de Salamanca, 1996.
- [NB91] W.Ñewman and M. Branicky. Real-time configuration space transforms for obstacle avoidance. *The International Journal of Robotics Research*, 6, 10 1991.

- [New87] W. Newman. High speed robot control and obstacle avoidance using dynamic potential functions. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 14–24, 1987.
- [NS83] C. Nash and S. Sen. *Topology and geometry for physicists*. Academic Press, London, 1983.
- [SD89] Z. Shiller and S. Dubowsky. Robot path planning with obstacles, actuators, gripper and payload constraints. *The International Journal of Robotics Research*, 8(6), 1989.
- [Sha87] M. Sharir. Efficient algorithms for planning purely translational collision-free motion in two and three dimensions. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1326–1331, 1987.
- [Spi82] M. Spivak. *Cálculo en variedades*. Reverté, Barcelona, 1982.
- [SV89] M. W. Spong and M. Vidyasagar. *Robot dynamics and control*. John Wiley and Sons, New York, 1989.
- [Udu77] S. Udupa. *Collision Detection and avoidance in controlled manipulators*. PhD thesis, Department of Electrical Engineering, California Institute of Technology, 1977.