

PIB-15912

**CENTRO DE INVESTIGACION Y DE ESTUDIOS AVANZADOS
DEL INSTITUTO POLITECNICO NACIONAL
UNIDAD GUADALAJARA**

**"SIMULADOR DE UN VEHICULO
AUTOGUIADO"**

TESIS QUE PRESENTA:

ING. FABIO FERNANDEZ RAMIREZ

**PARA OBTENER EL GRADO DE
MAESTRO EN CIENCIAS
EN LA ESPECIALIDAD DE
INGENIERIA ELECTRICA**

Guadalajara, Jal., 1998

**CINVESTAV I. P. M.
SECCION DE INFORMACION
Y DOCUMENTACION**

CLASIF.	
ADQUIS.	TESIS 1999
FECHA	ene. 22/99
PROCESO	Depto. de Servicios Bibliográficos
	\$

Bibliográficos

DIRECTOR DE TESIS
DR. ANTONIO RAMÍREZ TREVIÑO

Índice General

1	INTRODUCCIÓN.	3
1.1	DEFINICIÓN DE VEHICULO AUTOGUIADO.	3
1.2	¿CÓMO SURGEN LOS AGV?	4
1.3	CLASIFICACIÓN.	5
1.4	APLICACIONES.	6
1.5	METODOS DE GUIA Y NAVEGACIÓN.	9
1.6	FORMULACIÓN DEL PROBLEMA.	11
1.7	ORGANIZACIÓN DE LA TESIS.	12
2	DESCRIPCIÓN DEL SISTEMA.	13
2.1	DESCRIPCIÓN GENERAL DEL SISTEMA.	13
2.2	PLANTEAMIENTO DE LOS PROBLEMAS.	17
2.2.1	PROBLEMA BÁSICO. .	17
2.2.2	REPRESENTACIÓN DEL MAPA.	20
2.2.3	SELECCIÓN DE LA RUTA ÓPTIMA.	29
2.2.4	DETERMINACIÓN DE LA TRAYECTORIA.	32
2.2.5	SEGUIMIENTO DE LA TRAYECTORIA.	33
2.2.6	TAREAS REACTIVAS.	34
3	REPRESENTACIÓN DEL MAPA Y BÚSQUEDA DE LA RUTA ÓPTIMA.	37
3.1	REPRESENTACIÓN DEL MAPA.	38
3.1.1	MÉTODO DE DESCOMPOSICIÓN EXACTA DE CELDAS.	38
3.1.2	GENERACIÓN DE CELDAS.	41
3.1.3	GENERACIÓN DEL GRAFO DE CONECTIVIDAD.	44
3.2	BÚSQUEDA DE LA RUTA ÓPTIMA.	54
3.2.1	ALGORITMO A*	56
4	GENERACIÓN Y SEGUIMIENTO DE TRAYECTORIAS.	61
4.1	B-SPLINES.	61
4.2	PROPIEDADES DE LAS B-SPLINES.	65

4.3	MODELO DEL SISTEMA.	74
4.4	SEGUIMIENTO DE LA TRAYECTORIA B-SPLINE.	76
5	TAREAS REACTIVAS.	83
5.1	SENSORES.	83
5.2	DETECCIÓN DE OBSTÁCULOS EN LA TRAYECTORIA.	86
5.3	MODIFICACIÓN DE LA TRAYECTORIA.	90
6	CONCLUSIONES Y TRABAJO FUTURO.	101
APENDICE 1. DIAGRAMAS DE FLUJO DEL PROGRAMA SIMULADOR.		105
APENDICE 2. INSTRUCTIVO DE OPERACION DEL PROGRAMA SIMULADOR.		111
BIBLIOGRAFIA		113

Capítulo 1

INTRODUCCIÓN.

1.1 DEFINICIÓN DE VEHICULO AUTOGUIADO.

En nuestros días, el desarrollo de la ciencia y la tecnología exige una vertiginosa búsqueda de soluciones en todos los campos de investigación. Este desarrollo es impulsado en gran medida por las necesidades de un mundo cambiante, donde los requerimientos del mercado son cada día más exigentes y la competencia por ofrecer nuevos y mejores productos con un costo menor es el objetivo primordial de las empresas.

La búsqueda de costos bajos de producción ha originado la necesidad de automatizar equipos, sistemas, líneas de producción, y en general los sistemas de manufactura de las industrias y que esta sea una prioridad en nuestros días, teniendo como objetivo bajar los costos de los productos finales y poder competir en un mercado cada vez más exigente en precio y calidad.

Un reflejo de esta visión de la industria moderna, son las líneas automáticas de producción que cuentan con robots para una gran variedad de tareas y aplicaciones y que en la actualidad se pueden encontrar en la mayoría de las empresas en todo el mundo. Debido a los costos asociados en los procesos de manufactura por el manejo de inventarios y almacenes de productos terminados, los sistemas automáticos de movimiento, transporte de materiales y almacenamiento, continúan ganando una especial atención y aceptación en las industrias modernas. La amplia gama de industrias que necesitan del manejo de materiales y sistemas de almacenamiento, hacen que una parte de la comunidad científica enfoque su atención hacia los problemas que aparecen en los sistemas de manufactura, en el manejo, control, movimiento y almacenamiento de partes.

En la industria, los AGV (*Automated Guided Vehicles*) se presentan como una herramienta muy adecuada en el transporte, manejo y sistemas de almacenamiento de partes y piezas dentro de los sistemas de manufactura. Por su versatilidad, este tipo de vehículos son muy

importantes para muy diversas aplicaciones en diferentes tipos de industrias, en algunas de las cuales existen sistemas muy complejos de control de tránsito de vehículos, para formar todo un Sistema de Vehículos Autoguiados o AGVS.

A continuación se dará una definición de lo que se considera un AGV según el Materials Handling Institute [15]

Definición 1 *Un Vehículo Autoguiado es un vehículo de baterías sin conductor que puede ser programado para seleccionar y seguir una trayectoria y están equipados para seguir trayectorias cambiantes o variables.*

Un AGV de acuerdo a la definición anterior es un vehículo y como tal es un medio de transporte. Los AGV se emplean en el transporte de piezas o partes de producción como se ha mencionado, sin embargo es conveniente señalar cómo surgen, su clasificación, las principales aplicaciones que han tenido y las técnicas para guiar estos vehículos en una ruta.

1.2 ¿CÓMO SURGEN LOS AGV?

Según lo establecen algunos autores [15], los AGV surgen como tales en Estados Unidos a mediados de los años cincuenta, sin embargo estos fueron empleados inicialmente en Europa después de la Segunda Guerra Mundial como “tractores sin conductor” En los sesenta se implementaron los primeros vehículos sin conductor en la industria como equipos de transporte de grandes cargas y con muy poca flexibilidad en sus trayectorias. En los setenta el desarrollo en las computadoras afectó la tecnología de los AGV y los vehículos se convirtieron en bidireccionales, capaces de mover material entre las líneas de ensamble, naves industriales y estaciones de proceso.

A partir de entonces los AGV han tenido una evolución acelerada. En los ochenta se perfeccionó el sistema mecánico incorporando brazos robot y otros sistemas de autocarga y descarga, se trabajó en la programación de microprocesadores para controlar los vehículos, se instalaron diferentes tipos de sensores para asegurar la seguridad en las áreas de producción, etc. El desarrollo de los vehículos fue tal que propició una gran variedad en aplicaciones y creó la plataforma adecuada para formar igualmente, una gran variedad de tipos de vehículos.

En los noventa se ha trabajado principalmente en el Software de los sistemas de control con el fin de hacer vehículos más eficientes. Las aplicaciones han ido del plano estrictamente industrial a ambientes muy diversos incluyendo programas espaciales, proyectos automotrices y proyectos de investigación en los cuales se trabaja actualmente en diferentes centros de investigación a nivel mundial.

1.3 CLASIFICACIÓN.

Una clasificación general que se puede hacer de los AGV es la siguiente [11]:

- *Trenes sin Conductor.* En esta clasificación se incluyen los vehículos con remolques. Este tipo de vehículos pueden jalar uno, dos o más remolques, formando un tren de transporte de materiales. Son muy útiles para transportar grandes cargas en distancias largas.
- *Vehículos AGV Montacargas.* Son básicamente vehículos montacargas sin conductor. Dentro de este tipo de AGV existe una gran gama de vehículos para diferentes capacidades de carga y de alturas de torre. La mayoría de estos vehículos requieren de un operador humano para cargar y descargar el vehículo, sin embargo existen AGV totalmente automáticos que se encargan de cargar y almacenar las tarimas en estantes o gabinetes.
- *AGV transportadores de unidades de carga.* Este tipo de vehículos transportan unidades de carga de una estación a otra. Generalmente están equipados de sistemas automáticos de carga y descarga por medio de rodillos, bandas móviles, plataformas con pistones, brazos robot, u otros dispositivos. Dentro de este tipo de AGV se puede hacer una subdivisión:
 - *AGV de Carga Ligera.* Mueven cargas pequeñas o partes simples.
 - *AGV de Líneas de Ensamble.* Son diseñados para transportar partes a lo largo de una línea de ensamble por las diferentes estaciones de trabajo donde se van armando las piezas transportadas por el AGV.
- *AGV sin carga.* En este tipo de vehículos se incluyen los vehículos que se emplean para diferentes aplicaciones donde la función básica del AGV no es el transporte de piezas o cargas. En este tipo se incluyen los vehículos de los programas espaciales y diferentes programas de investigación, etc.



Figura 1.1: AGV transportador de contenedores de 15 m. y 40 ton. 150 vehículos operan en la terminal ECT/Sea-Land Delta, Rotterdam. Emplean cables guía con una longitud total de 250 Km en sus trayectorias.

1.4 APLICACIONES.

Las principales razones que tienen las industrias manufactureras para invertir en sistemas de transporte automatizados varían de acuerdo a las ventajas que les ofrezcan en sus aplicaciones.

Los AGV han sido usados desde mediados de siglo, como ya se ha mencionado. Estos vehículos varían en un amplio rango de tamaños para transportar desde pequeñas cargas de unos cuantos kilogramos hasta cargas de más de 100 toneladas (ver la figura 1.1). El ambiente de trabajo de los AGV puede variar desde oficinas alfombradas hasta ambientes agresivos al aire libre.

Los AGV modernos son controlados por computadora con microprocesadores a bordo de ellos. La mayoría de los AGVS cuentan con un centro de control por computadoras, este centro de control se encarga de optimizar la utilización de los AGV del sistema, dando las órdenes de transporte, monitoreando el seguimiento de la transferencia de materiales y dirigiendo el tráfico de los AGV.

Muchos AGVS emplean además sistemas de carga y descarga automatizados. En estos sistemas el AGV puede cargar y descargar "pallets" o cargas automáticamente usando sistemas de brazos robot, bandas, etc. Esto depende del tipo y tamaño de las unidades de carga que manejan. Los primeros AGV eran remolques, trenes con varios remolques o vehículos en forma de plataformas, cuya única función era el transporte de las cargas. Los AGV modernos pueden estar equipados con brazos robot y tenazas que realizan funciones de robots para tomar piezas de carga y descarga (ver la figura 1.2). Otros AGV están equipados con sistemas montacargas, este tipo de vehículos manejan cargas en almacenes con estantes y son capaces de levantar las cargas hasta 10 metros o más (ver la figura 1.3).

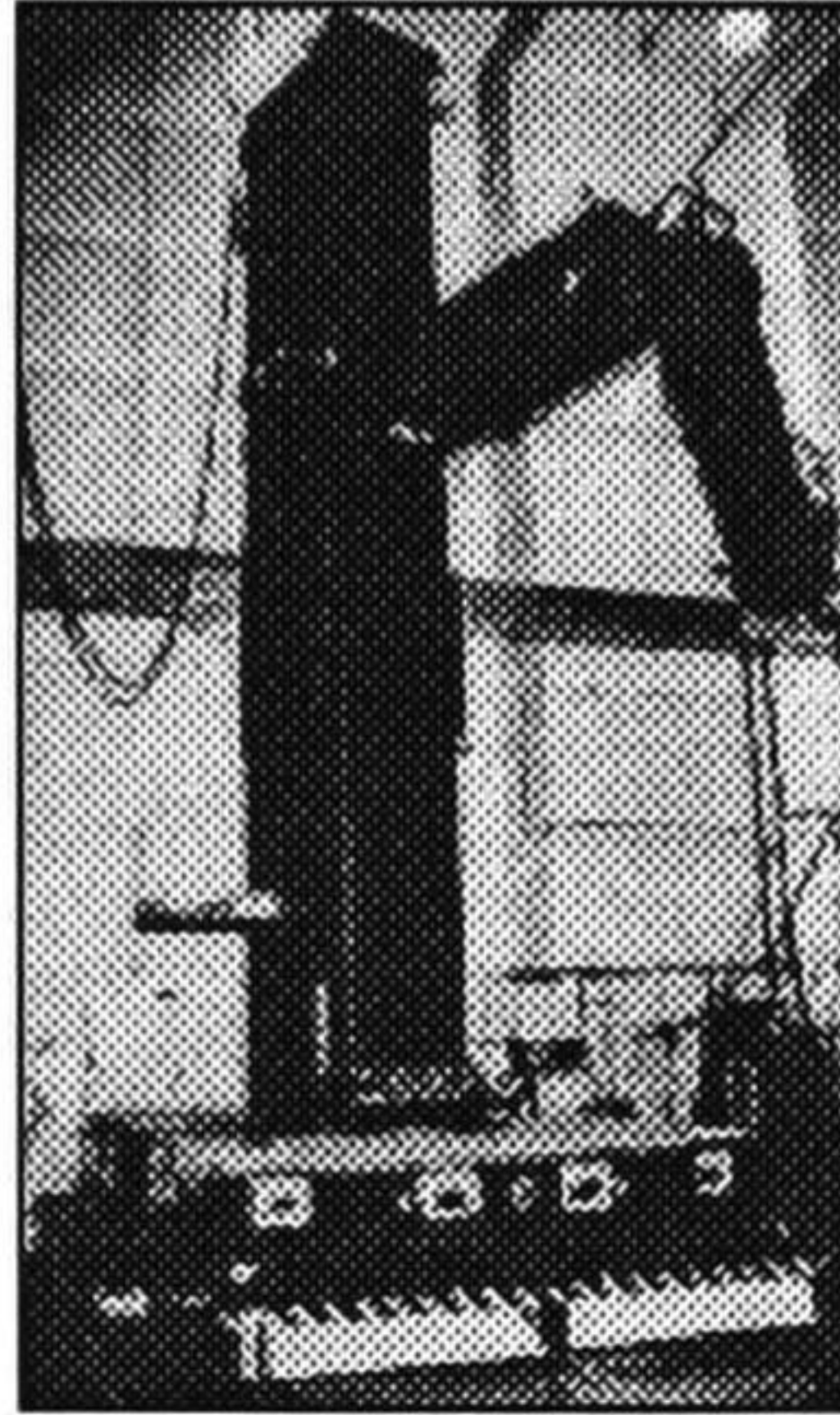


Figura 1.2: AGV diseñado por United Technologies (UTC) y Pratt & Whitney Waterjet Systems, para transportar el robot industrial más grande del mundo.

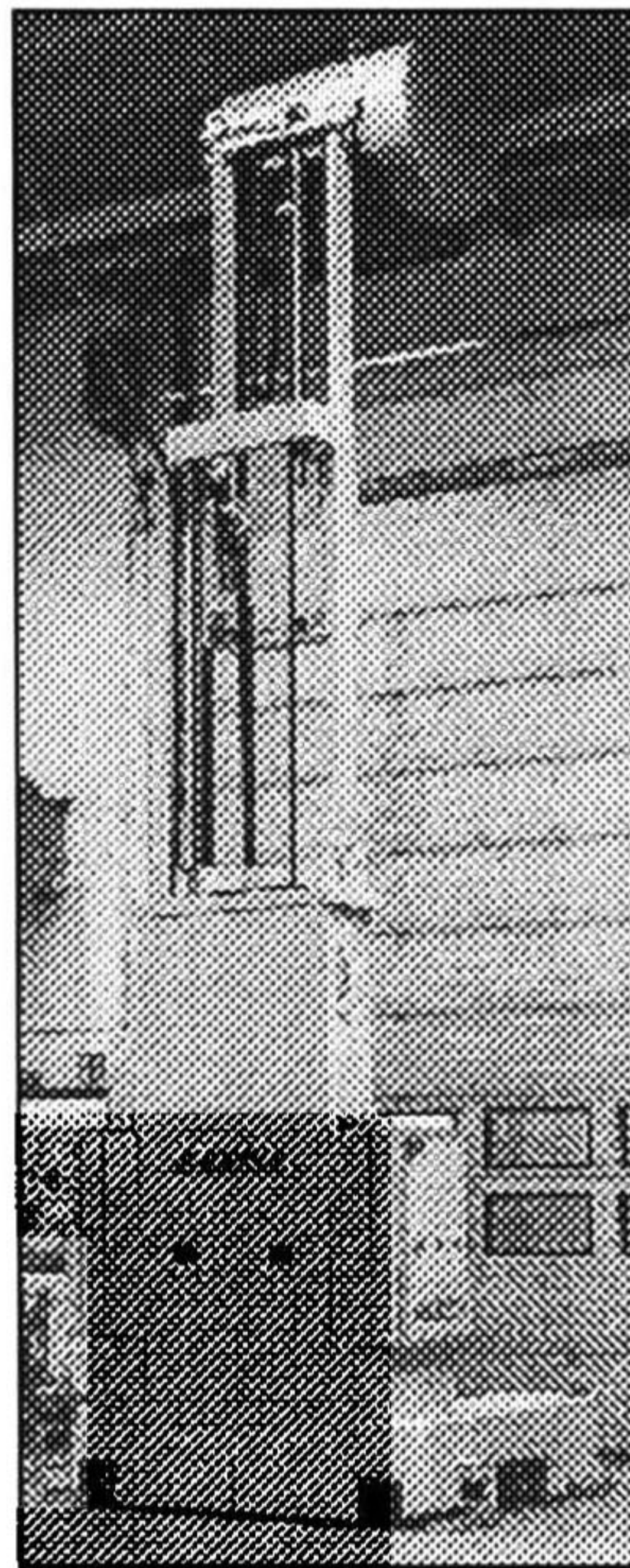


Figura 1.3: AGV-LFQ Montacargas con torre de altura de 7 mt. Han sido instalados dos en Bemis/Mactac, Bélgica, para manejar rollos de papel y plástico.

Las aplicaciones que tienen los AGV actualmente se pueden dividir en:

1. *Sistemas de transporte de cargas.* Esta es la primera aplicación que tuvieron los AGV y en esta categoría se incluyen aquellas aplicaciones donde el AGV cumple una función de medio de transporte de cargas. Aquí se incluyen los trenes sin conductor y algunos AGV transportadores de unidades de carga.

2. *Sistemas de distribución y almacenamiento.* Esta es una de las principales aplicaciones industriales de los AGV donde se han tenido las mayores ventajas. Más del 70% de los AGV industriales se emplean para esta tarea. Los AGV transportadores de unidades de carga y los AGV montacargas son utilizados comunmente para estas aplicaciones.

3. *Operaciones en líneas de ensamble.* Se emplean los AGV de Líneas de Ensamble que son diseñados especialmente para esta función. Este tipo de vehículos se emplean en líneas donde la tasa de producción es relativamente baja (entre 4 y 10 minutos por estación en la línea) y debido a diferentes modelos o productos de la línea este tipo de transporte de los ensambles le dan gran flexibilidad al sistema de producción.

4. *Sistemas de Manufactura Flexible (SMF).* En los SMF los AGV se emplean como los sistemas de manejo de materiales. Los vehículos transportan los materiales de las estaciones de carga a las diferentes estaciones de trabajo en el sistema. Los vehículos además transportan las piezas entre estaciones de trabajo en el sistema de manufactura. En una estación de trabajo, las piezas son transferidas de los AGV al área de trabajo de la estación para ser procesadas, al terminar el ciclo de trabajo de la estación regresa un AGV para llevar las piezas a la siguiente estación. Los AGV proporcionan un sistema de manejo de materiales muy versátil para complementar la flexibilidad en la operación de los SMF.

5. *Aplicaciones misceláneas.* Incluyen aquellas aplicaciones no incluidas en un ambiente de manufactura ni almacenes. Tales aplicaciones son muy variadas, por ejemplo, como entrega de correo en oficinas (ver la figura 1.4), operaciones de manejo de materiales en hospitales y más recientemente en los programas espaciales donde se ha apoyado con grandes inversiones la investigación en este campo. Además algunas compañías automotrices están trabajando actualmente en proyectos de investigación para tratar de llevar algunas de las características de los AGV a los automóviles.

Como se puede apreciar las expectativas en este campo son muy amplias y es por ello que se debe apoyar la investigación encaminada en este sentido. No obstante que los AGV son equipos ampliamente utilizados en la industria mundial, se siguen buscando nuevos y mejores algoritmos de control y ruteo para hacer vehículos cada vez más eficientes e “inteligentes”

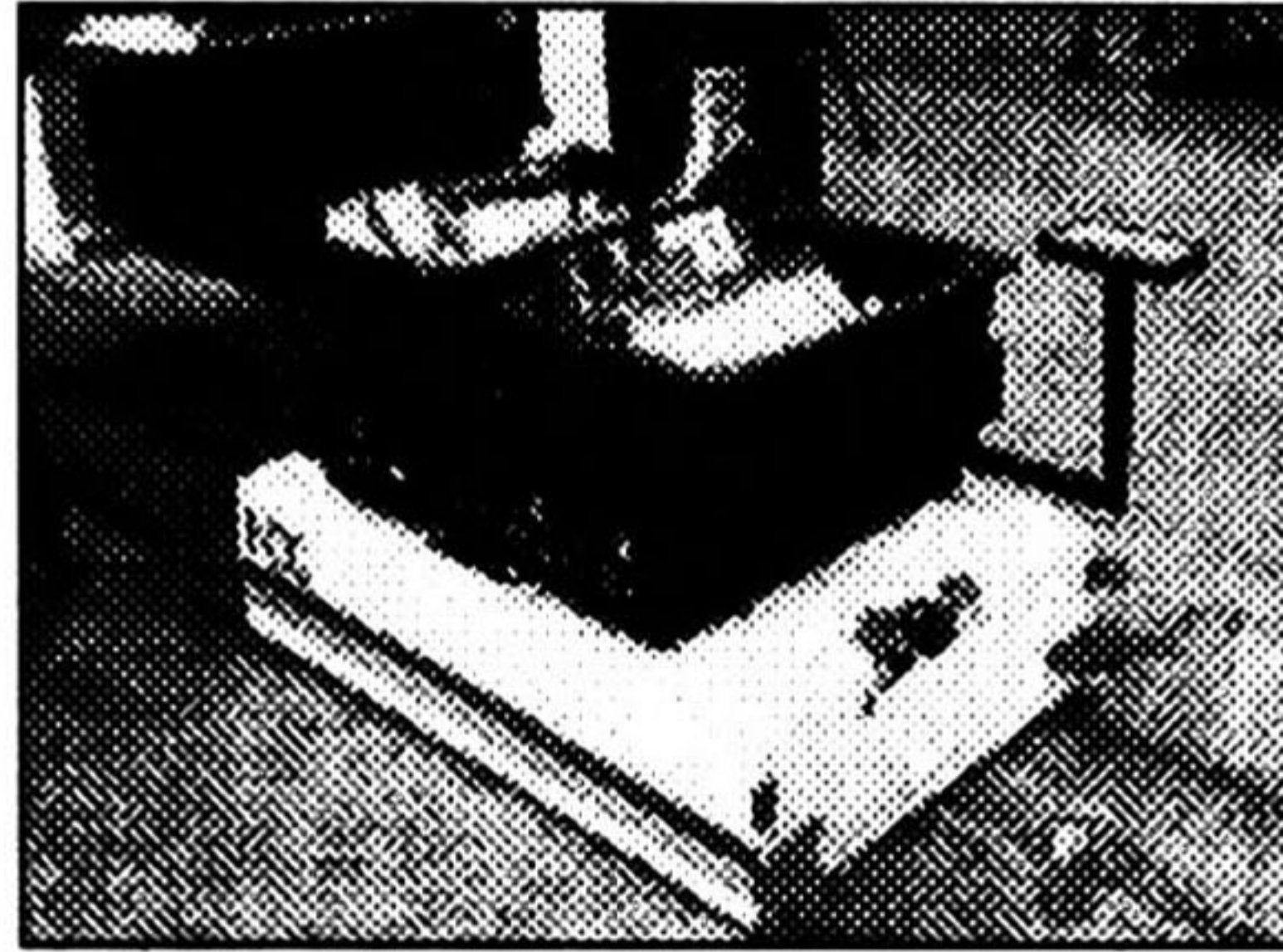


Figura 1.4: AGV Mail-Deliver empleado para entregar correo en un ambiente de oficinas.

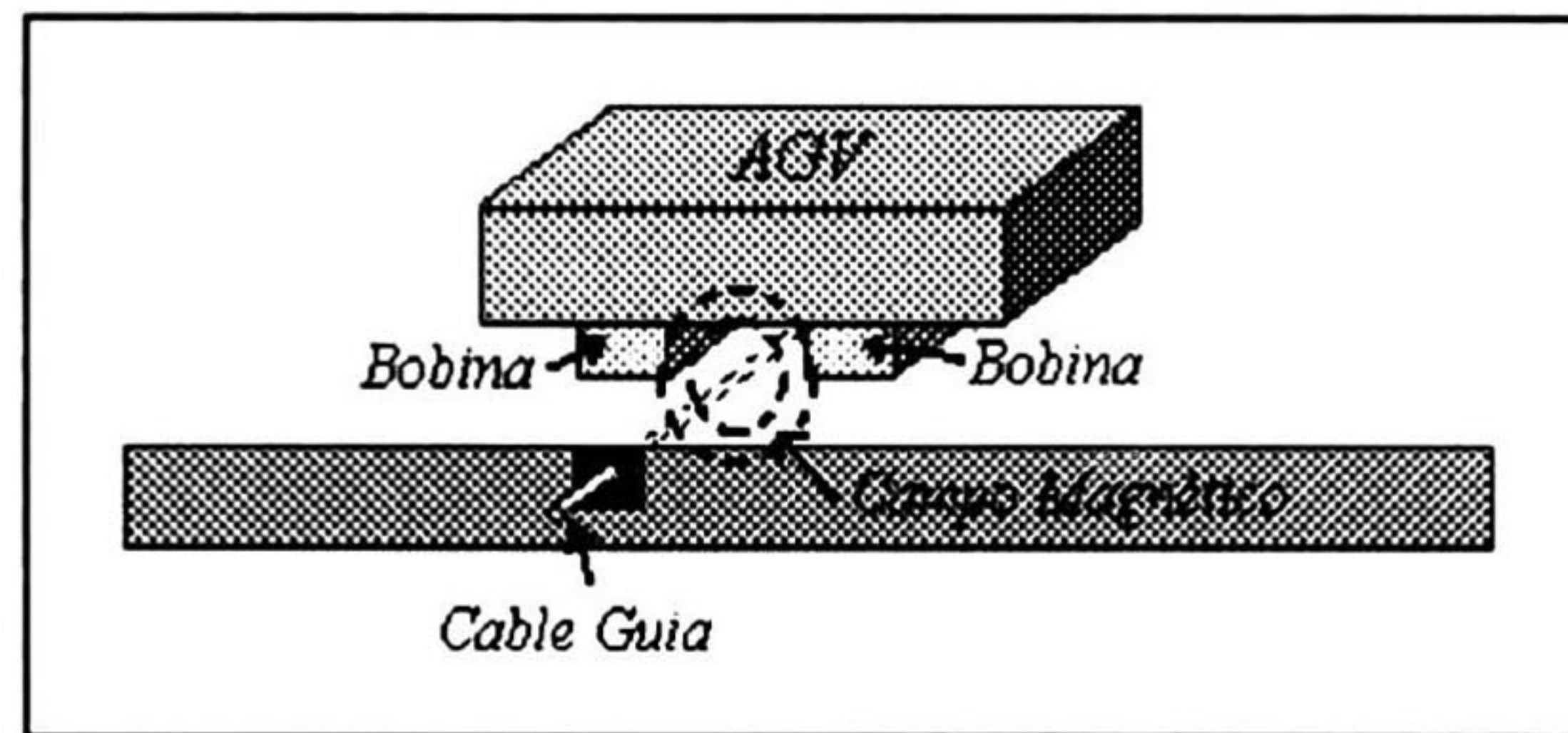


Figura 1.5: El sistema de cable guía emplea un sistema de antena para mantener al AGV dentro de la trayectoria del cable.

1.5 METODOS DE GUIA Y NAVEGACIÓN.

Se pueden implementar diferentes métodos de guía y navegación en los AGV. Estos métodos se refieren a la forma en que se definen o se determinan las trayectorias de los AGV y los sistemas de control del vehículo empleados para seguir estas trayectorias.

Los primeros AGV seguían como guía, un cable inductivo enterrado en el piso o una línea óptica visible pintada o hecha con una cinta sobre el piso. La guía de cable inductivo sigue siendo el sistema de guía más usado para los AGV que viajan sobre pisos de concreto, aún en nuevas instalaciones (ver la figura 1.5).

Muchas compañías se han dedicado a buscar una manera de sustituir las guías de cable inductivo. En los últimos años los sistemas de guía y navegación han implementado técnicas como los "scanners" de láser, señales de microondas, sensores de ultrasonido, imanes enterrados, sistemas de visión con cámaras, etc.

Las principales técnicas de guía son las siguientes, [15, 11]:

Tow line. Es un tipo primitivo de carro jalado por una cadena o cable.

Guía de Cable. Vehículos con sensores tipo antena que siguen un cable energizado enterrado en el piso.

Guía Inercial. Usan una tarjeta con un microprocesador en el vehículo para dirigir al vehículo en una trayectoria preprogramada; emplea sensores de ultrasonido, proximidad o algún otro tipo de sensor sonar para la detección de obstáculos y realizar un cambio direccional.

Infrarojo. Se transmite una luz infraroja y se refleja desde reflectores instalados en el techo, detectores tipo radar transmiten la información a la computadora para que calcule distancias y determine la posición e indique la dirección de movimiento.

Láser. El láser busca reflectores con códigos de barras montados en las paredes. A través de distancias conocidas y de las mediciones de distancia que han recorrido las ruedas del vehículo, el AGV puede maniobrar con precisión.

Óptica. Fotosensores leen y siguen partículas fluorescentes pintadas en concreto o de cintas pegadas en cualquier tipo de piso.

Tipo Enseñanza. Se enseñan las trayectorias y se graban; es decir, se lleva al vehículo por la trayectoria deseada y este prácticamente aprende el camino e indica a la computadora el nuevo camino aprendido.

Algunos de los nuevos tipos de sistemas de guía para los AGV son llamados *sistemas de libre rango de navegación*. A los vehículos que emplean este tipo de sistemas se les denomina *vehículos inteligentes*. Sin embargo todos los sistemas de guía requieren instalar referencias a lo largo de las trayectorias para recalibrar los AGV en el trayecto, así que ningún sistema es cien por ciento libre o totalmente "inteligente". Algunos sistemas modernos de cables guías son usados sólo como referencias estáticas y el AGV puede moverse fuera de los cables guía. La diferencia entre todos los sistemas de navegación es el grado de libertad que se le da al AGV en el sistema y siempre va relacionado con el costo, la complejidad de los algoritmos de control, los costos en cambios del programa de control y el mantenimiento. El problema es que mientras se tiene mayor grado de libertad el software es más complejo y se requiere un especialista para hacer algún cambio en él.

Algunos AGV modernos emplean radiocomunicaciones comunes, con un radio FM para transmitir datos de control hacia los vehículos y estos a su vez transmitir información de su posición, dirección y velocidad, que sirve para la retroalimentación del control y su monitoreo. Las radiocomunicaciones en sí mismas han proporcionado un alto grado de libertad en el diseño de los nuevos sistemas de AGV.

En el mercado existen a la venta diferentes tipos de AGV. En algunos casos, cuando se vende un AGV se vende también un simulador del AGV. Este simulador sólo tiene la finalidad de comprobar la trayectoria que seguirá. Los algoritmos de control no pueden ser cambiados por el usuario ya que el controlador es fijo para el AGV vendido (la información sobre AGV, simuladores y cursos de entrenamiento está disponible por medio de las compañías que los venden, por ejemplo AGV mentor).

Sin embargo, para ambientes de investigación este tipo de simuladores no es adecuado, ya que no sólo se quiere ver si el AGV realizará la tarea satisfactoriamente, sino también poder

cambiar los diferentes algoritmos de control que posee el AGV. El contar con un simulador de un AGV ayuda a diseñar los algoritmos de control y a obtener muchos errores antes de que sean realmente implementados. Como se verá posteriormente, un AGV consta de varias partes, como son: parte mecánica, sensores, actuadores y control (el cual a su vez se compone de varios niveles de control). Un buen simulador debería contar con un modelo de cada una de estas partes e integrarlo como un todo. Además el usuario debe poder, a voluntad, cambiar un modelo. Así, se pueden probar diferentes combinaciones de algoritmos de control, y más aún, si el AGV cambia, es posible cambiar su modelo dentro del simulador.

1.6 FORMULACIÓN DEL PROBLEMA.

El problema a estudiar en esta tesis es el diseño de un AGV (cuyo modelo será mostrado en el capítulo 4) de arquitectura abierta. Es decir un AGV que permita estudiar e implementar en él diferentes algoritmos de representación del entorno de trabajo y de control.

El objetivo es el estudio de diferentes algoritmos de representación del entorno y de control y seleccionar una arquitectura (incluye la parte mecánica, la electrónica y algoritmos de control) que pueda ser utilizada en la investigación de vehículos autoguiados.

Como resultado de esta tesis se tiene el estudio de los diferentes algoritmos y la selección de la arquitectura y la implementación de un simulador de AGV.

La arquitectura se describe sucintamente a continuación. La parte mecánica es una plataforma de forma cilíndrica con dos ruedas motorizadas y una libre, como se muestra en el capítulo 4. El mapa de trabajo consiste en una vista de plafón. Para obtener el espacio libre se utiliza un algoritmo de descomposición de celdas, el cual permite la generación del grafo de conectividad (los puntos que puede alcanzar el AGV); el algoritmo A* determina la mejor ruta a seguir. Finalmente la ruta es suavizada utilizando curvas B-spline. Existen un módulo para introducir el modelo de los sensores y otro de tareas reactivas que permiten evadir obstáculos durante el recorrido.

El otro punto importante es la elaboración de un simulador que permite simular la arquitectura seleccionada, pero que además es abierto en el sentido de que es posible sustituir cualquier algoritmo que conforman esta arquitectura (los algoritmos son los modelos de la parte mecánica, de los sensores, de la navegación y de control local). Actualmente el simulador tiene implementadas todas las partes descritas en la arquitectura, a excepción de la generación de celdas. La complejidad del algoritmo de generación de celdas es polinomial [13], la del A* es, en general, no polinomial [19]. Afortunadamente estos algoritmos trabajan fuera de línea. Los algoritmos de control son polinomiales, la suavización de la ruta se realiza en línea, pero su complejidad lineal, por lo que los algoritmos del simulador pueden ser tomados para trabajar en tiempo real.

En esta tesis se toman como antecedentes los resultados obtenidos en [25], que propone el empleo de curvas B-splines en el control de un vehículo móvil y se realiza su implementación en el programa simulador. La tesis se complementa con un método de descomposición de celdas del espacio libre del vehículo y de representación del conocimiento del mapa mediante grafos, mejorando el método de descomposición trapezoidal descrito en [13], en la búsqueda

de la ruta óptima se implementó en el simulador un algoritmo A* [19, 14]. Además, se propone una solución en la evasión de obstáculos modificando la trayectoria al detectarse un obstáculo y ésta es implementada en las tareas reactivas del simulador.

1.7 ORGANIZACIÓN DE LA TESIS.

En el capítulo 2 se describen las partes que componen a un AGV, se plantean los problemas de representación de conocimiento del mapa del espacio de trabajo del AGV, ruteo, seguimiento de trayectorias, búsqueda de la trayectoria óptima y las soluciones propuestas para cada uno de ellos. Se plantea, en general, una visión global del sistema, los problemas involucrados y las soluciones propuestas para cada uno de ellos.

El capítulo 3 describe los algoritmos de representación del conocimiento del mapa y de la búsqueda de la ruta óptima. Estos métodos son los que emplea el simulador para representar el mapa del espacio de trabajo, y la manera en que se emplea esta representación para hacer una búsqueda de la ruta óptima en distancia, para ir de un punto a otro dentro del espacio de trabajo del AGV.

En el capítulo 4 se presenta la forma en que, a partir de la ruta previamente determinada, se establece la trayectoria que va a seguir el AGV. También se presenta la manera en que se realiza el control para el seguimiento de la trayectoria definida. Aquí se incluye el modelo matemático del AGV que es empleado en el programa simulador para implementar en él los algoritmos de control realizados.

En el capítulo 5 se describe la forma en que el vehículo realizará las tareas reactivas, esto es, el algoritmo que se sigue para evadir obstáculos que se presenten en el seguimiento de la trayectoria definida. En el programa se puede simular un obstáculo en movimiento durante el seguimiento de la trayectoria del AGV.

Finalmente en el capítulo 6 se dan las conclusiones obtenidas en este trabajo y el trabajo futuro en este proyecto.

En los apéndices se encuentran los diagramas de flujo de todos los algoritmos programados en el simulador y una guía para utilizar el software del simulador implementado.

Capítulo 2

DESCRIPCIÓN DEL SISTEMA.

2.1 DESCRIPCIÓN GENERAL DEL SISTEMA.

En este capítulo se dará un panorama general de las partes que componen a un AGV. Es importante puntualizar los problemas generales que se deben resolver para lograr que un AGV cumpla con su función. En el capítulo anterior se explicó que existe una clasificación amplia y que los AGV son herramientas que se emplean en muy diversas aplicaciones. Aquí explicaremos las características principales que tienen los AGV, planteando los problemas que se deben resolver. Estos problemas son generales para cualquier tipo de AGV, y para cada aplicación se tienen diversos esquemas de solución. Es decir, que existen diversas maneras de resolver los problemas dependiendo de la aplicación propia del vehículo. El tipo de solución que se le da a cada uno de estos problemas va muy ligada a los costos y al presupuesto que se tenga.

La manera más adecuada de resolver los problemas que se presentan en el diseño del “Hardware” y “Software” de los AGV depende de las necesidades y recursos del cliente, así como del criterio del fabricante o diseñador. En este capítulo se da también, una explicación general de las soluciones tomadas en el desarrollo del proyecto para los problemas planteados. En los capítulos posteriores se discuten con mayor detalle estas soluciones y los algoritmos implementados en el modelo del vehículo del simulador.

Es necesario tener una visión global del sistema de un AGV, para poder descomponerlo en bloques. Al hacer una división del sistema por tareas, se pueden formar módulos de funciones muy específicas. La primer división del sistema es muy general y se muestra en la figura 2.1. Para explicar las tareas que debe realizar un AGV, de acuerdo a esta división, debemos dar las siguientes definiciones [13], que serán utilizadas a lo largo de esta tesis:

Definición 2 Sea A un objeto rígido simple (el AGV) en movimiento dentro de un espacio Euclideo W , representado como R^N , con $N=2$ ó 3 . Y sean B_1, \dots, B_n objetos rígidos fijos dentro de W . El espacio W es llamado el espacio de trabajo y los n objetos son llamados obstáculos.

Definición 3 Sean F_A y F_W ejes Cartesianos puestos en A y W respectivamente. F_A son ejes en movimiento, mientras que F_W son ejes fijos. Cada punto a de A tiene una posición fija con respecto a F_A pero tiene una posición en W que depende de la posición y orientación de F_A relativos a F_W . Entonces, una configuración q de A se define como una especificación de la posición η y orientación Θ de F_A con respecto a F_W .

Definición 4 El espacio de configuración de A se define como el espacio C de todas las configuraciones de A .

Definición 5 Cada obstáculo B_i , $i=1, \dots, n$, en el espacio de trabajo W se mapea en C en una región: $CB_i = \{q \in C \mid A(q) \cap B_i \neq \emptyset\}$ que es llamada C-obstáculo.

Definición 6 La unión de todos los C-obstáculos: $\cup_{i=1}^n CB_i$ es llamada Región C-obstáculo.

Definición 7 El conjunto: $C_f = C \setminus \cup_{i=1}^n CB_i = \{q \in C \mid A(q) \cap (\cup_{i=1}^n B_i) = \emptyset\}$ es llamado espacio libre.

Definición 8 Cualquier configuración en C_f es llamada configuración libre.

Definición 9 Una trayectoria de A desde la configuración q_{ini} hasta la configuración q_{fin} es un mapeo continuo: $\eta : [0,1] \rightarrow C$. Donde $\eta(0) = q_{ini}$ y $\eta(1) = q_{fin}$.

Definición 10 Una trayectoria libre entre dos configuraciones libres q_{ini} y q_{fin} es un mapeo $\eta : [0,1] \rightarrow C_f$, con $\eta(0) = q_{ini}$ y $\eta(1) = q_{fin}$.

Definición 11 *Una ruta es una serie finita de posiciones en C_f*

NOTA: En el desarrollo siguiente de la tesis nos referiremos a la trayectoria libre como trayectoria, a menos que se haga alguna aclaración.

La figura 2.1 muestra, como se mencionó anteriormente, una descomposición general del sistema básico de un AGV; en él se especifican los módulos fundamentales siguientes:

Planeador de la Ruta. Primeramente se debe definir la ruta que seguirá el AGV; la ruta puede ser fija o variable de acuerdo a la aplicación específica del AGV. En aplicaciones como transporte de cargas y en líneas de ensamble se prefiere tener rutas fijas pre-establecidas, por cuestiones de seguridad de las cargas y las personas. En cambio en aplicaciones como transporte de piezas en los SMF las rutas son preferentemente variables, controladas por una central de control para adecuarse a las rutas requeridas para cubrir las necesidades del SMF. En un número considerable de aplicaciones se prefiere una combinación de rutas fijas con cierta libertad de variación de la misma durante el recorrido, esto le proporciona seguridad al sistema con cierta flexibilidad, tal es el caso de los sistemas con AGV Montacargas. Las rutas que siguen los AGV dentro de su espacio de trabajo pueden ser determinadas por el diseñador del sistema, una central de control o el propio vehículo.

Planeador de la trayectoria. Una vez especificada la ruta que debe cubrir el AGV en su recorrido, se debe establecer la trayectoria que se seguirá. El planeador de la trayectoria es en sí, el generador de la trayectoria del AGV en función del tiempo. Existen trayectorias que son pre-definidas, las generadas durante el recorrido, y una combinación de ambas. El hecho de que la ruta sea fija no implica que la trayectoria haya sido pre-definida. Esto se puede observar en algunos sistemas con cables enterrados en los cuales las rutas son fijas, sin embargo los vehículos pueden desviarse del cable guía para evadir algún obstáculo y volver nuevamente a recuperar la trayectoria. La trayectoria es generada por un planeador que se encuentra preferentemente en las centrales de control de los AGVS o en el mismo vehículo.

Controlador del Seguimiento de Trayectorias. Esta es la etapa de control de los AGV en la cual se va a seguir la trayectoria deseada. El control actúa directamente sobre los actuadores con el fin de poder desviar el AGV de la configuración actual q_s , determinada con ayuda de los sensores, hacia la configuración de la trayectoria q_d deseada. El tipo de control que se emplea para esta función es muy variable y depende de factores tales como el tipo de actuadores del vehículo, el tipo de sensores empleados, el desempeño propio del controlador, el diseñador o programador, etc.

Actuadores del Vehículo. En este bloque se consideran las acciones propias de los actuadores, el movimiento que producen los mismos y la configuración alcanzada con sus movimientos, velocidades, fuerzas, etc. Los sensores permiten calcular de manera directa o indirecta la configuración del AGV y retroalimentar la información necesaria al controlador del seguimiento de trayectorias para rectificar la entrada en los actuadores y lograr el seguimiento de la trayectoria.

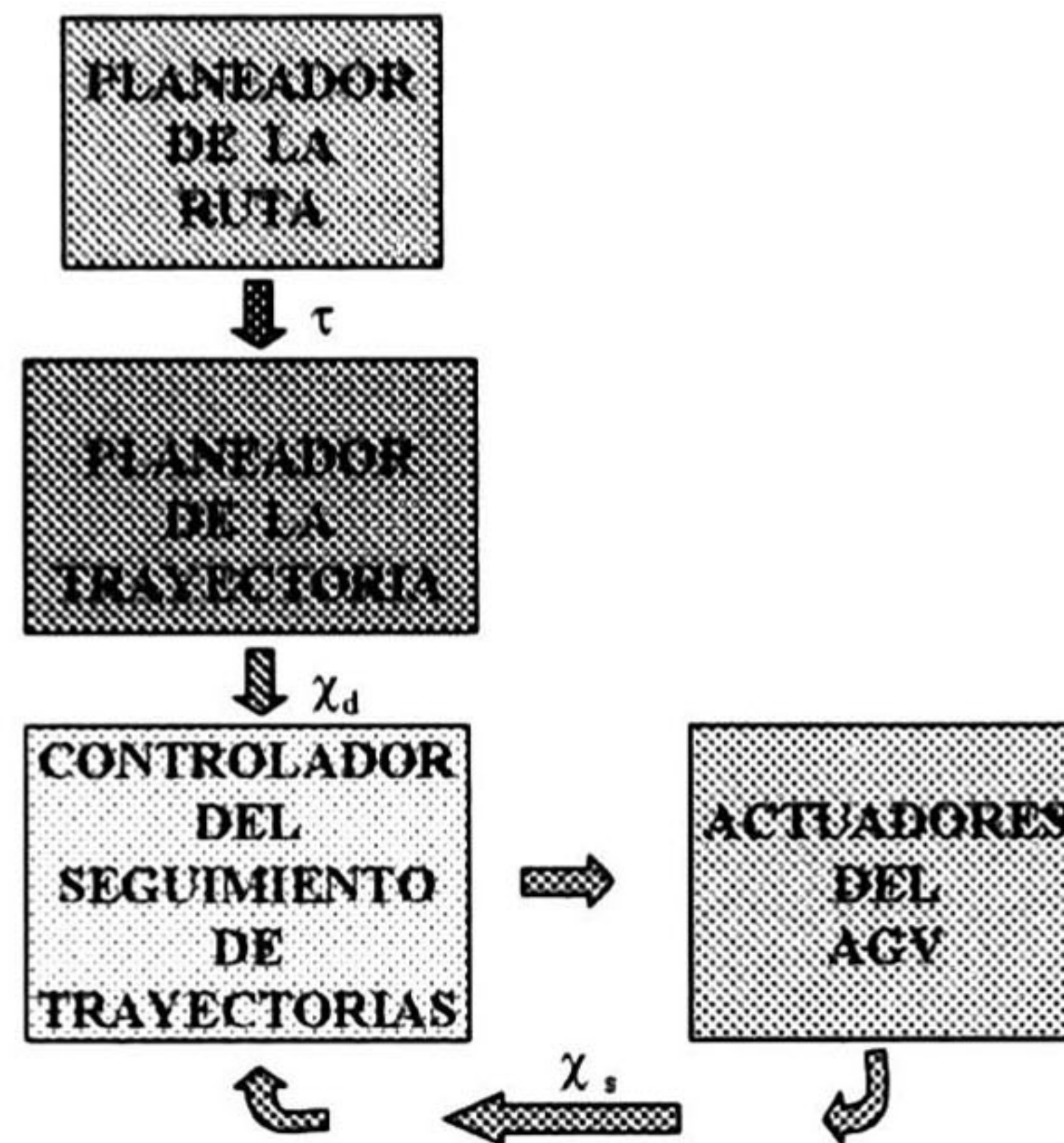


Figura 2.1: La ruta τ es producida por el Planeador de Rutas y el Planeador de la Trayectoria calcula la trayectoria deseada χ_d . La trayectoria χ_s es medida por los sensores y es la información de retroalimentación.

Los cuatro bloques anteriores forman parte, como se ha mencionado, de las tareas básicas de los AGV. Existen, para diferentes aplicaciones, diseños de AGV con funciones más complejas que se complementan con otros bloques como pueden ser: sistemas de alarmas, sistemas de carga y descarga automática, monitoreo de la configuración en centrales de control, controles de tráfico en los AGVS, evasión de obstáculos durante la trayectoria, etc.

En el simulador desarrollado en esta tesis se emplea un planeador de la ruta y un planeador de trayectorias a bordo del vehículo; este último genera la trayectoria durante el recorrido. Estos planeadores son dos algoritmos independientes que forman un control general de navegación. Se emplean también controles PID para controlar la velocidad de dos motores que producen el movimiento en la ruedas de tracción del vehículo y poder realizar el seguimiento de la trayectoria con la información obtenida por unos sensores de velocidad de las ruedas. Para el control de la trayectoria se emplea el modelo matemático del vehículo al cual se aplica el control con los datos de velocidad de los motores. Se consideran además sensores de ultrasonido para la detección de obstáculos con el fin de evadirlos durante el recorrido de la trayectoria. Los sensores fueron simulados de acuerdo a observaciones del funcionamiento de sensores de ultrasonido reales realizadas por experimentación. La figura 2.2 muestra el esquema del sistema considerado. En los capítulos 3, 4 y 5 se especifican con más detalle todas estas características del AGV simulado y se presentan los algoritmos, modelos, pruebas y resultados obtenidos.

2.2 PLANTEAMIENTO DE LOS PROBLEMAS.

2.2.1 PROBLEMA BÁSICO.

Como se verá en esta sección los problemas relativos al movimiento de los AGV dentro de su espacio de trabajo son diversos. Estos problemas se resuelven de manera modular, lo que hace que al dividir el problema básico en varios problemas, éstos se puedan resolver por separado para formar en conjunto una solución al problema básico.

Para definir el problema básico asumiremos que el AGV es el único objeto móvil en el espacio de trabajo. Además limitaremos los movimientos a movimientos sin contacto, de manera que los efectos relacionados a la interacción mecánica por el contacto físico entre objetos se puede ignorar. Asumiendo además que el AGV es un objeto simple rígido, es decir, un objeto cuyos puntos son fijos con respecto a cada uno de los demás, los movimientos de este objeto son sólo limitados por los obstáculos.

Bajo las consideraciones anteriores el problema básico de los AGV se puede definir como el problema básico de los robots [13]:

Definición 12 *Dada una posición y orientación inicial, y una posición y orientación final de A en W , el problema básico es generar una trayectoria χ especificando una secuencia continua de posiciones y orientaciones de A evitando el contacto con los B_i 's, empezando en la posición y orientación iniciales, y terminando en la posición y orientación finales, reportando una falla si tal trayectoria no existe.*

El problema básico implica realizar los módulos fundamentales descritos en la sección anterior. La división modular del problema básico depende del tipo de aplicación y el tipo de AGV que se diseñe. Así pues, aquellos vehículos que tienen rutas y trayectorias fijas son más sencillos y sus limitaciones de diseño permiten esquemas menos complicados para resolver satisfactoriamente el problema básico. Por ejemplo, un AGV con restricciones en sus movimientos, es decir, poca o nula capacidad de modificar su trayectoria y rutas; al detectar un obstáculo se debe detener y activar una alarma. Por el contrario, los vehículos inteligentes deben ser capaces de activar sus tareas reactivas que le permitan esquivar los obstáculos modificando la trayectoria original y retomándola posteriormente, con lo que sus algoritmos de control y navegación se complican considerablemente. Con esto, es claro que el tipo de problemas que se deben resolver para que los vehículos cumplan con su función dependen en sí del tipo de vehículo y de su respectiva aplicación.

En la figura 2.2 se presentan los bloques principales que componen a un AGV. Los problemas principales que se deben resolver para el diseño, construcción e implementación del AGV se pueden agrupar en:

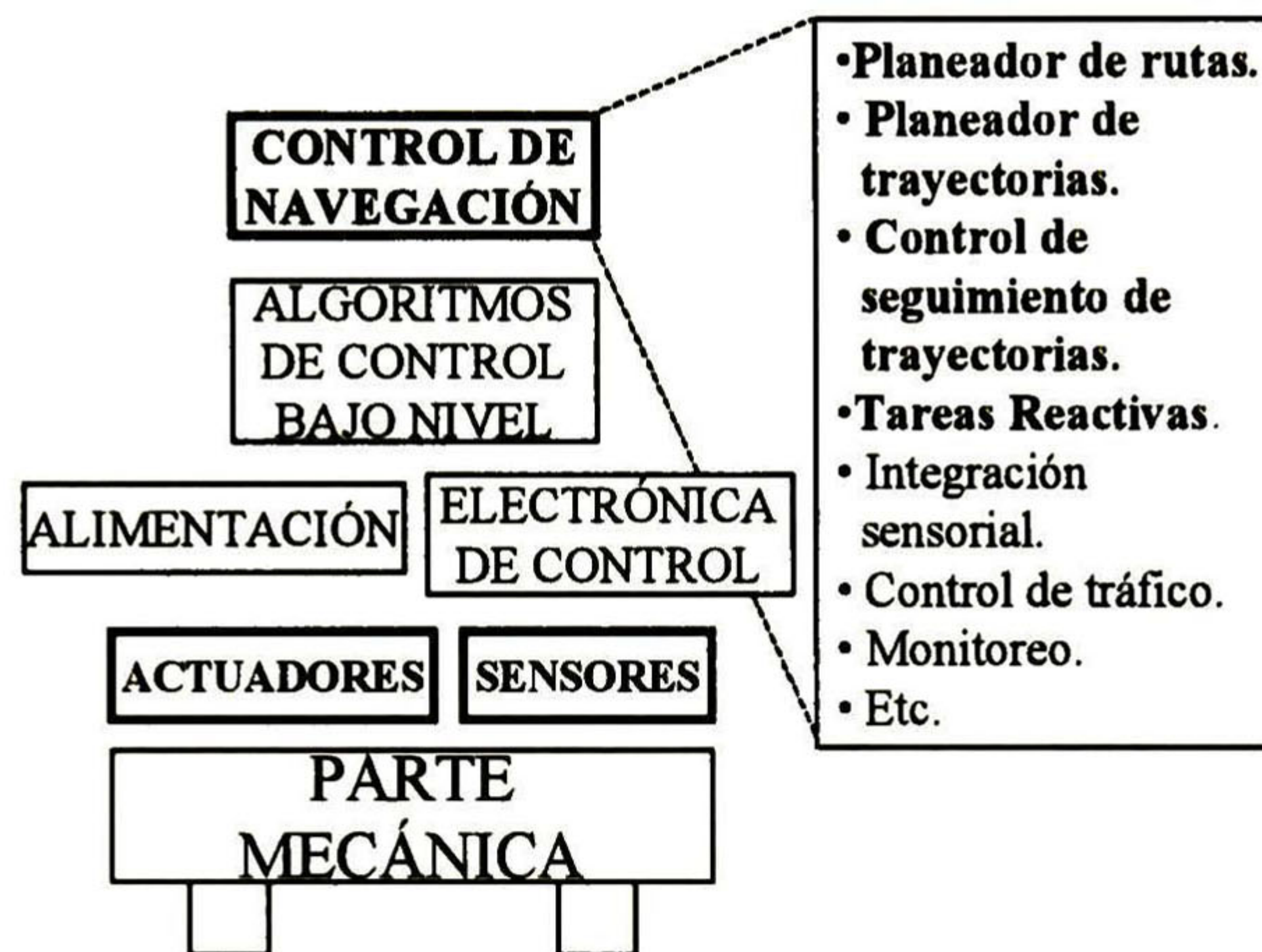


Figura 2.2: Diagrama de bloques del sistema de un AGV.

1. “*HARDWARE*”. Aquí se incluyen todos aquellos problemas involucrados con la parte mecánica, eléctrica y electrónica del vehículo. *El reto que se tiene es crear un vehículo capaz de realizar las tareas que se le asignan en su trabajo y tener un costo de operación bajo, siendo un vehículo seguro para la carga que transporta y las personas en su ambiente de trabajo.* El “hardware” para un AGV se compone de los siguientes elementos:

- *Diseño del vehículo (materiales, forma y distribución de los dispositivos).*
- *Fuente de alimentación.*
- *Sistema de tracción.*
- *Electrónica de potencia para los “drivers” de los motores.*
- *Sistema de carga de baterías.*
- *Sensores y su electrónica de control.*
- *Alarmas.*
- *Alimentación para sensores, alarmas, y dispositivos de electrónica.*
- *Sistema de carga y descarga automático (si se requiere).*
- *Otros.*

2. “*SOFTWARE*”. Incluye todos aquellos problemas que se relacionan con los algoritmos que controlan el vehículo. *El reto es controlar la operación del vehículo y sus diferentes*

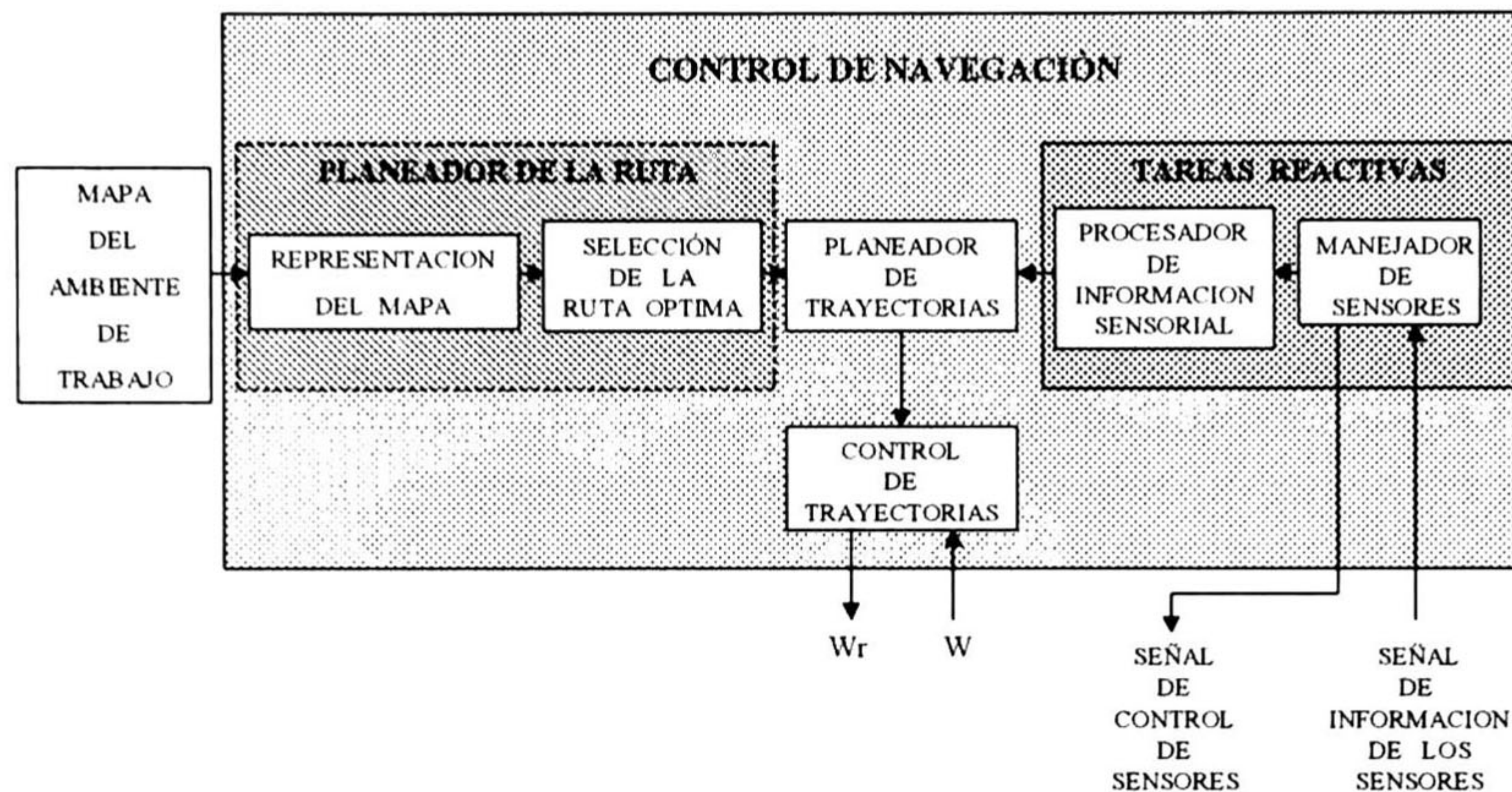


Figura 2.3: Sistema de Control de Navegación del AGV del simulador.

dispositivos. El “software” de un vehículo inteligente se puede agrupar en los algoritmos siguientes:

- *Representación del Mapa.*
- *Selección de la Ruta Óptima.*
- *Determinación de la Trayectoria.*
- *Seguimiento de la Trayectoria.*
- *Tareas Reactivas.*
- *Manejador de sensores. (Control e Integración Sensorial).*
- *Control de Tráfico. (En AGVS).*
- *Otros.*

Los problemas en los que se enfoca el estudio de esta tesis son relacionados con el *software* de un vehículo inteligente, que corresponden a los algoritmos del control de navegación resaltados en la figura 2.2. Estos algoritmos se relacionan entre sí como lo muestra la figura 2.3. A continuación en esta sección, se muestran varias formas de visualizar estos problemas y se hace mención de algunos métodos de solución.

2.2.2 REPRESENTACIÓN DEL MAPA.

Este problema se presenta en el caso de que el vehículo o una central de control tengan que definir las trayectorias. Esto implica la necesidad de tener información que permita conocer el espacio de trabajo del AGV y poder tomar decisiones en la planeación del movimiento del vehículo. La cantidad de información que se le pueda dar a priori al vehículo o a la central de control sobre el entorno del AGV determina la facilidad con que pueda elegir la trayectoria adecuada en el menor tiempo.

Si se tienen rutas o trayectorias fijas predeterminadas este problema prácticamente no existe o se soluciona con algoritmos muy sencillos. Sin embargo, donde no se tienen rutas fijas se requiere tener conocimiento del espacio de trabajo en que se mueve el AGV. Es fundamental representar la información del espacio de trabajo. La representación del mapa entonces, debe ser de una manera adecuada para que el sistema la pueda utilizar en la toma de decisiones.

A pesar de tantas diferencias, los métodos de planeación del movimiento se basan en unos cuantos métodos generales de aproximación. Entre estos, los más importantes son: *Mapa de caminos, descomposición de celdas y campo potencial*. A continuación se describirán brevemente estos métodos con el fin de presentar algunas de las diferentes opciones en la solución de este problema [13, 9].

Mapa de Caminos.

El método de Mapa de Caminos de planeación de movimiento consiste en determinar la conectividad del espacio libre del robot en una red de curvas unidimensionales, llamada el *mapa de caminos*, que está sobre el espacio libre C_f o su cerradura $cl(C_f)$.

Una vez que se construye el mapa de caminos \mathcal{R} , se emplea como un conjunto de trayectorias normalizadas. Entonces la planeación de las trayectorias se reduce en encontrar la trayectoria que conecte las configuraciones inicial y final con puntos en \mathcal{R} y hacer una búsqueda en \mathcal{R} para conectar estos puntos. La trayectoria encontrada, si existe, es la concatenación de tres subtrayectorias: una subtrayectoria que conecta la configuración inicial a un punto en \mathcal{R} , otra subtrayectoria contenida en \mathcal{R} , y una última subtrayectoria conectando a \mathcal{R} con la configuración final.

Como mencionamos anteriormente este método es una idea general, y de ella se desprenden varios métodos particulares que emplean esta idea. Entre estos métodos se encuentran: el *grafo de visibilidad*, el *diagrama de Voronoi*, la *red de camino libre y silueta*. Dos de estos métodos se describen a continuación, después de dar algunas definiciones, [10, 14,19], relacionadas con la teoría de grafos que se emplearán en las explicaciones siguientes.

Definición 13 Un grafo G es un conjunto finito no vacío $V(G)$ de objetos llamados nodos (también llamados vértices o puntos) y un conjunto $E(G)$, posiblemente vacío, de subconjuntos de dos elementos de $V(G)$ llamados arcos (o líneas). El conjunto $V(G)$ es llamado conjunto vértice de G y $E(G)$ es su conjunto arco.

Definición 14 Sean u y v elementos de $V(G)$, si $e=\{u,v\}$ es un elemento de $E(G)$, representado por $e=uv$ o $e=vu$, entonces se dice que u y v son adyacentes en G , y que e une (o liga) a u y v .

Definición 15 El número de nodos en un grafo es llamado su orden, y el número de arcos es su tamaño.

Definición 16 Un grafo etiquetado tiene uno o más descriptores (o etiquetas) que corresponden a cada nodo para distinguirlo de cualquier otro nodo en el grafo.

Definición 17 Un grafo dirigido es aquel cuyos arcos tienen una dirección determinada. En un grafo dirigido el arco uv indica que la dirección del arco es de u a v y es diferente al arco vu , que indica lo contrario. Al recorrer un grafo dirigido se debe respetar la direccionalidad de los arcos.

Definición 18 Un grafo no dirigido es aquel en el que los arcos no tienen una dirección asociada y el arco uv y vu son el mismo. En los grafos no dirigidos no hay restricción en la dirección al recorrer el grafo.

Definición 19 Una secuencia ordenada de nodos $[n_1, n_2, n_3, \dots, n_n]$, donde cada par $\{n_i, n_{i+1}\}$ en la secuencia representa un arco (n_i, n_{i+1}) , es llamada trayectoria de longitud $n-1$ en el grafo.

Definición 20 Dos nodos en un grafo se dice que son conexos si existe una trayectoria en el grafo que los incluye a ambos.

Definición 21 Un grafo H es un subgrafo del grafo G si $V(H) \subseteq V(G)$ y $E(H) \subseteq E(G)$.

Definición 22 El peso $w(e)$ del arco $e=uv$ es un número asociado con e que corresponde a una representación de un costo entre los nodos u y v . El peso $w(e)$ puede representar un costo en tiempo, distancia o cualquier variable que se debe cubrir para poder pasar de u a v .

Definición 23 Un grafo con peso es aquel cuyos arcos tienen peso.

Definición 24 Un grafo con raíz tiene un nodo único, llamado raíz, tal que existe una trayectoria desde el nodo raíz hacia todos los demás nodos del grafo.

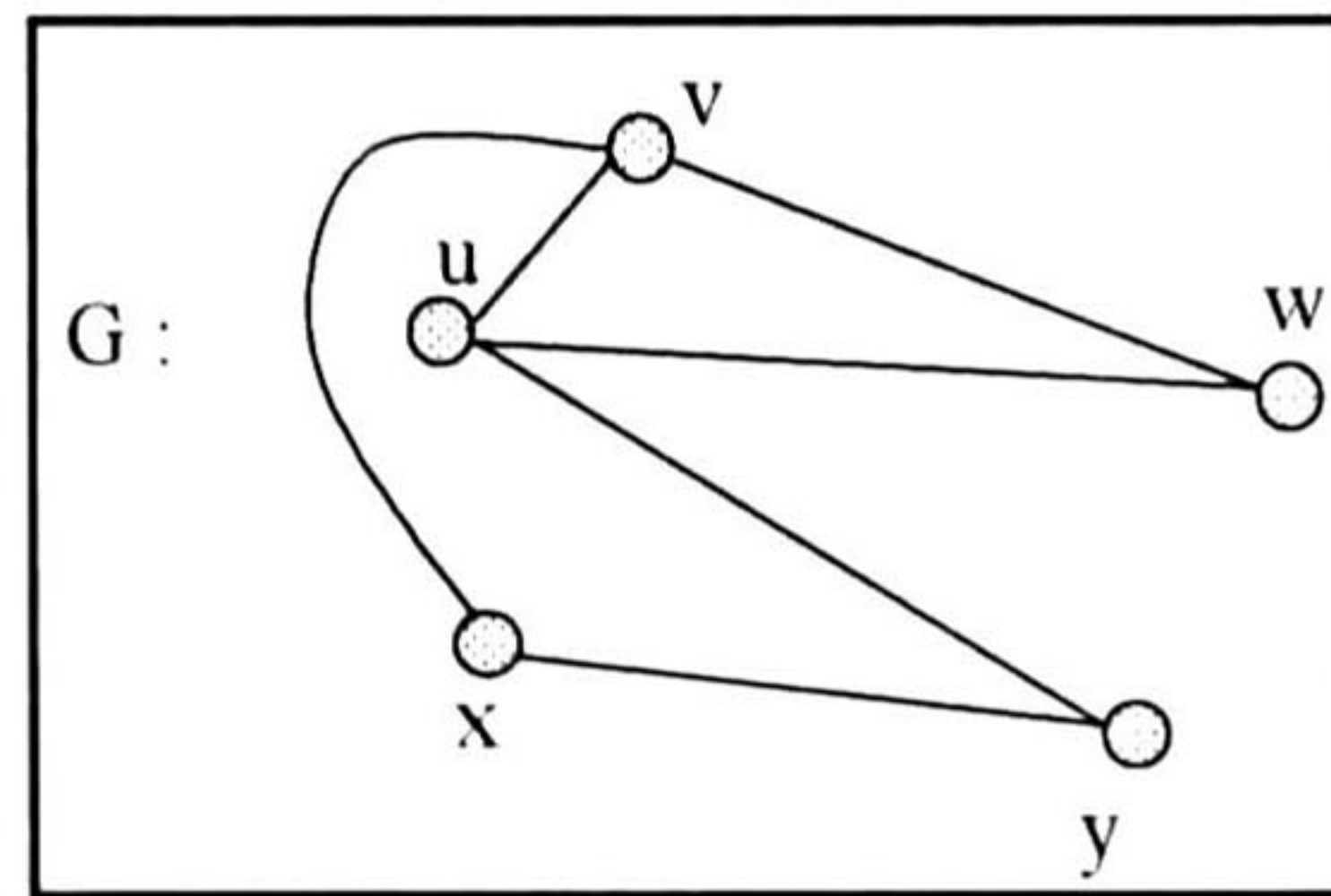


Figura 2.4: El grafo G es un grafo no dirigido sin peso en sus arcos.

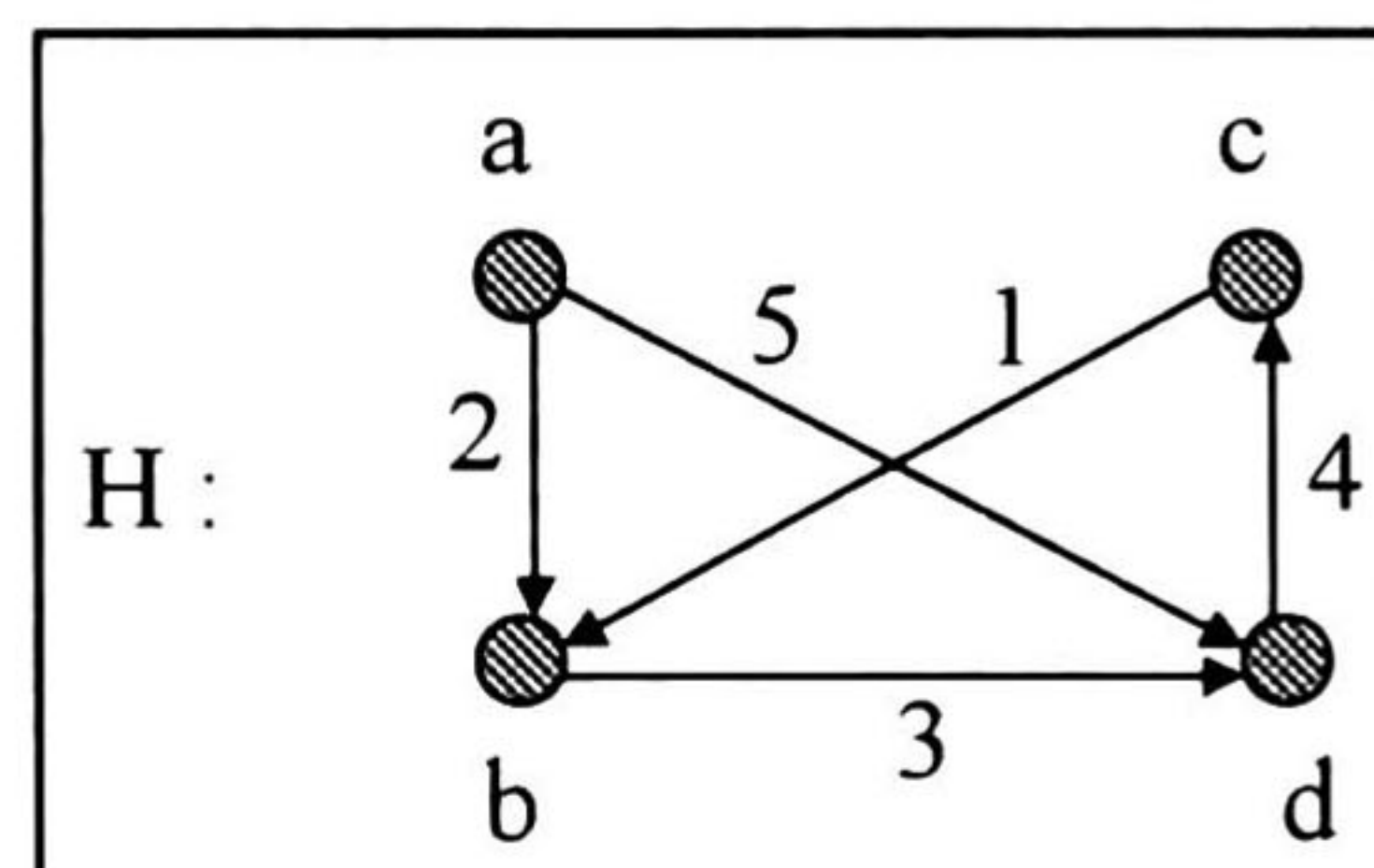


Figura 2.5: Grafo dirigido ponderado.

Cada grafo tiene un diagrama asociado, los nodos se representan por puntos o círculos y los arcos se representan por líneas o curvas que unen dos nodos adyacentes. Generalmente se hace referencia al diagrama del grafo como el grafo en sí. La figura 2.4 muestra un grafo G etiquetado no dirigido, con: $V(G)=\{u,v,w,x,y\}$ y $E(G)=\{uv,vw,wu,xv,xy,uy\}$. La figura 2.5 representa un grafo H etiquetado dirigido con peso, con: $V(H)=\{a,b,c,d\}$, $E(G)=\{ab,cb,bd,ad,dc\}$, $w(ab)=2$, $w(cb)=1$, $w(bd)=3$, $w(ad)=5$ y $w(dc)=4$. En el desarrollo del simulador de esta tesis se emplean grafos etiquetados no dirigidos con peso.

Una vez establecidas las definiciones anteriores se explicarán a continuación los métodos de grafo de visibilidad y el diagrama de Voronoi.

El *método del grafo de visibilidad* es uno de los primeros métodos aplicados. Este método se aplica principalmente a espacios de configuración bi-dimensionales con una región C-obstáculo poligonal. El grafo de visibilidad es el grafo no dirigido \mathcal{G} cuyos nodos son las configuraciones inicial q_{ini} y final q_{fin} , y todos los vértices de la región C-obstáculo. Los arcos de \mathcal{G} son todos los segmentos de línea recta que conectan dos nodos que no intersectan el interior de la región C-obstáculo. Los arcos del grafo también incluyen los contornos o lados de los C-obstáculos. La figura 2.6 muestra un ejemplo de un grafo de visibilidad. Los arcos del subgrafo de \mathcal{G} restringido a los vértices de la región C-obstáculo determinan el mapa de caminos \mathcal{R} . Los otros arcos de \mathcal{G} conectan las configuraciones inicial y final con \mathcal{R} .

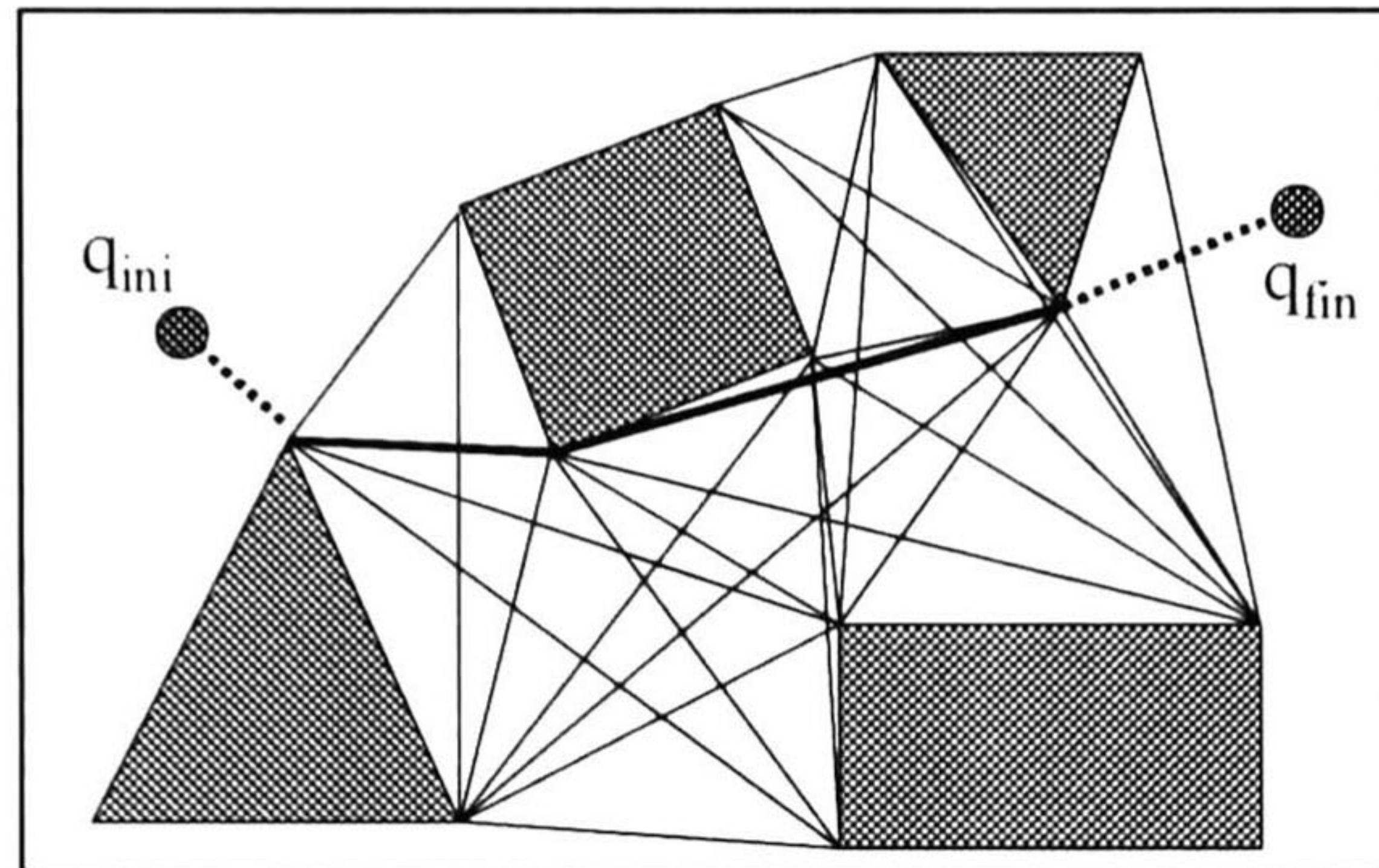


Figura 2.6: Grafo de visibilidad en un espacio de configuración bi-dimensional con C-obstáculos poligonales.

En \mathcal{G} se puede hacer una búsqueda de la trayectoria semi-libre más corta entre q_{ini} y q_{fin} , de acuerdo a la métrica Euclideana en \mathbf{R}^2 . Esta trayectoria, si existe, es una línea poligonal que conecta a q_{ini} y q_{fin} a través de los vértices de C-obstáculo.

El *método del diagrama de Voronoi*, llamado también *retracción*, consiste en definir una función continua de C_f dentro de un conjunto uni-dimensional de si mismo (el mapa) tal que la restricción de esta función a este subconjunto es el mapa de identidad (en Topología a tal función se le llama "retracción"). En una configuración con espacio bi-dimensional, la retracción de C_f es típicamente su diagrama de Voronoi. Este diagrama es el conjunto de todas las configuraciones libres cuya mínima distancia a la región C-obstáculo \mathcal{CB} es alcanzada con al menos dos puntos en los límites de \mathcal{CB} (ver la figura 2.7). La ventaja de este diagrama es que proporciona trayectorias libres las cuales tienden a maximizar la distancia entre el robot y los obstáculos. Cuando los C-obstáculos son polígonos, el diagrama de Voronoi consiste de segmentos rectos y parabólicos.

La manera de obtener el diagrama de Voronoi es calculando la distancia más corta de cada punto del espacio libre al obstáculo más cercano. Si esta distancia se interpreta sobre la figura 2.7 como una altura saliendo de la página, considerando una altura cero en los obstáculos y sus límites y aumentando conforme nos alejamos de ellos, la superficie tendrá líneas de las elevaciones formadas por puntos que son equidistantes a dos obstáculos o más. El diagrama de Voronoi consiste de estas líneas.

Las configuraciones inicial y final se retractan en el diagrama a q'_{ini} y q'_{fin} como se muestra en la figura 2.7. Una trayectoria es posteriormente buscada en el diagrama entre q'_{ini} y q'_{fin} . La trayectoria libre entre q_{ini} y q_{fin} producida por este método, si existe, consiste en tres subtrayectorias: la trayectoria recta entre q_{ini} y q'_{ini} , una trayectoria en el diagrama de q'_{ini} a q'_{fin} , y la trayectoria recta de q'_{fin} a q_{fin} .

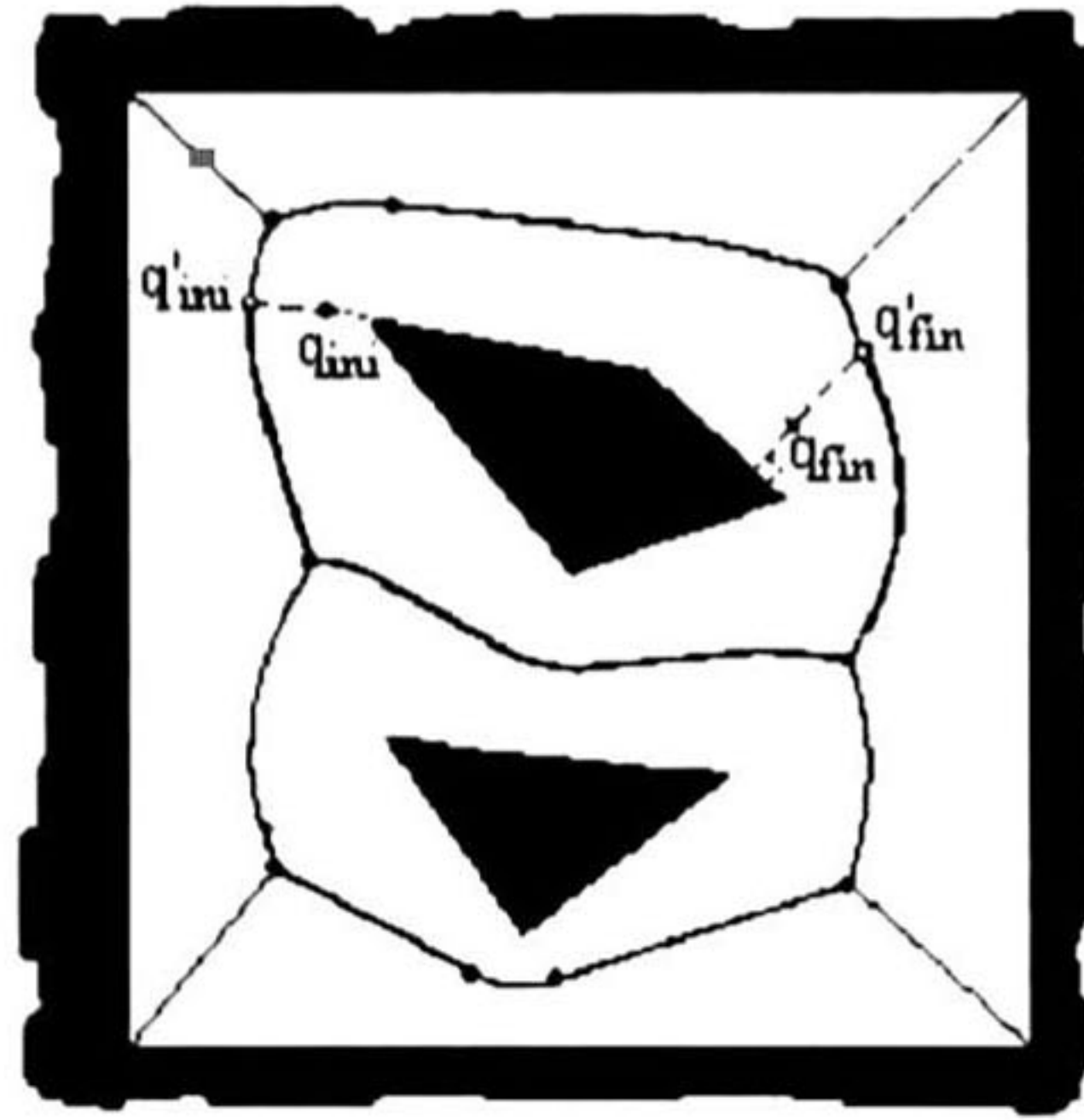


Figura 2.7: Diagrama de Voronoi en un espacio de configuración con una región C-obstáculo poligonal.

DESCOMPOSICIÓN DE CELDAS.

El método de descomposición de celdas consiste en descomponer el espacio libre del robot en regiones simples, llamadas celdas, de tal manera que una trayectoria entre dos configuraciones en una celda puede ser fácilmente generada. Un grafo no dirigido representando la relación de adyacencia entre las celdas es construido para generar en él las búsquedas. Este grafo es llamado el *grafo de conectividad*. Sus nodos son las celdas extraídas de C_f y dos nodos son conectados por un arco si y solo si, estos nodos corresponden a celdas adyacentes. El resultado de la búsqueda en este grafo es una secuencia de celdas llamada un *canal*. Este canal es el que llamaremos *ruta*. Así pues, una trayectoria continua se puede determinar a partir de esta ruta.

El método de descomposición de celdas se puede dividir en dos tipos, la *exacta* y la *aproximada*:

- Los métodos de descomposición exacta de celdas descomponen el espacio libre en celdas cuya unión es exactamente¹ el espacio libre. Los límites de la celda corresponden a un cambio crítico en el espacio libre de acuerdo a algún criterio de ordenamiento, es decir, un cambio repentino en las restricciones aplicadas al movimiento del robot.
- Los métodos de descomposición aproximada de celdas producen celdas de forma predefinidos, por ejemplo rectángulos o cuadrados, cuya unión y distribución corresponde estrictamente al cubrimiento del espacio libre. Los límites de una celda no caracterizan una discontinuidad de algún ordenamiento y no tienen un significado físico.

¹El término *exacto* se refiere a la forma en que se divide el espacio libre del robot, es decir a la descripción del mapa, no a la solución del problema físico.

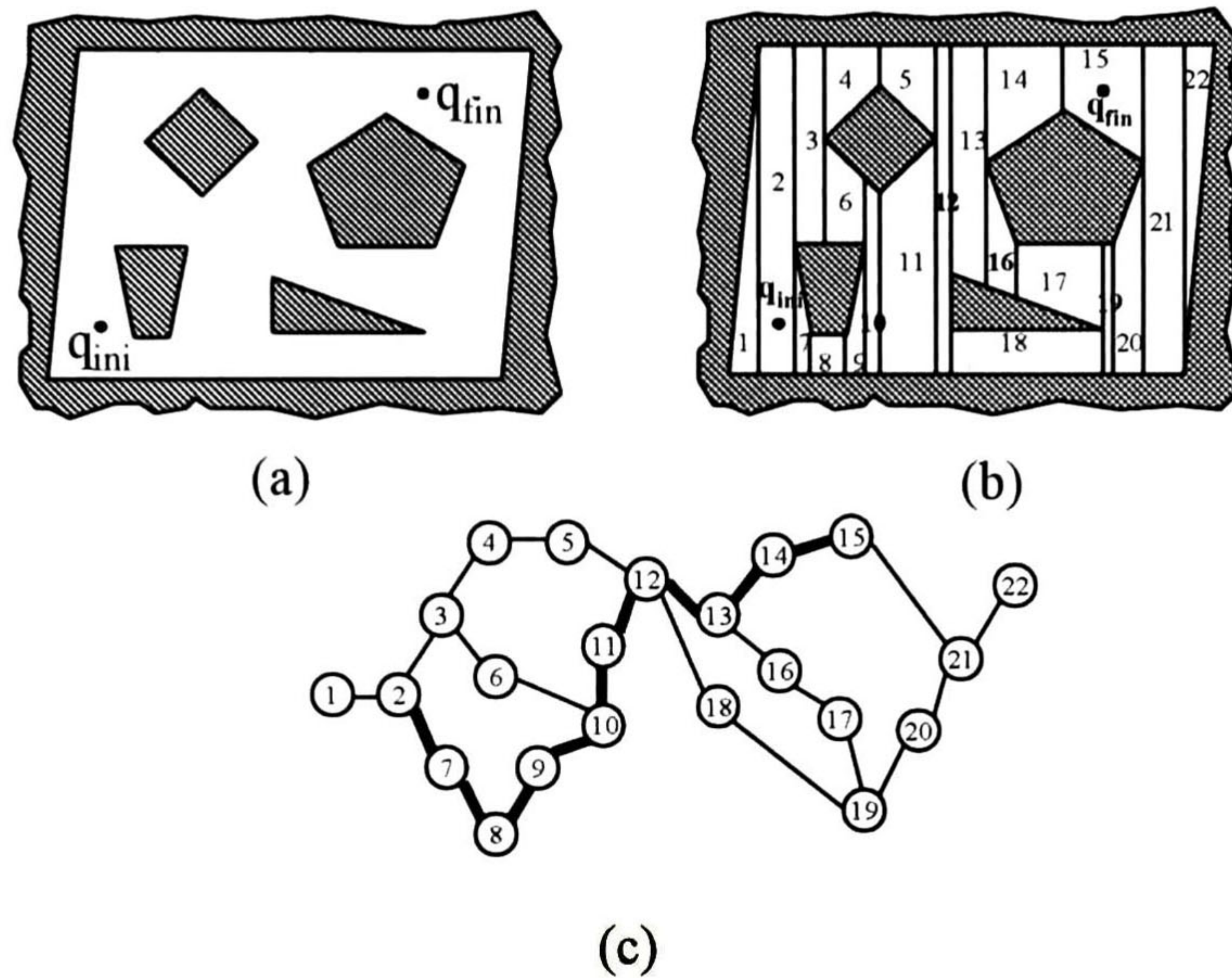


Figura 2.8: Descomposición exacta de celdas. Espacio libre limitado por una área de obstáculos poligonales (figura a). Se realiza la descomposición del espacio libre en celdas triangulares y trapezoidales (figura b). Con las celdas formadas se construye el grafo de conectividad donde se busca la ruta (figura c).

Las figuras 2.8 y 2.9 muestran una descomposición exacta de celdas en una configuración de espacio bi-dimensional. El espacio libre es limitado externamente por un polígono e internamente por cuatro polígonos. El espacio libre, como se puede observar se descompone exactamente en celdas trapezoidales y triangulares. Las celdas son construídas, en este caso trazando líneas verticales sobre C_f de los vértices de C -obstáculo, terminando al interceptar algún punto en C -obstáculo. Dos celdas son adyacentes si comparten una porción de longitud no-cero de una línea de la celdas. Apartir de la descomposición de celdas se puede crear el grafo de conectividad y encontrar en él una ruta, de esta manera se puede determinar una trayectoria que conecte las configuraciones inicial y final. Este método en particular es el empleado en la solución del problema básico en el simulador desarrollado y se verá con mayor detalle en el Capítulo 3.

La figura 2.10 muestra una descomposición aproximada de celdas. El espacio libre está limitado externamente por un rectángulo R e internamente por tres polígonos. El rectángulo R es recursivamente descompuesto o dividido en rectángulos más pequeños. Cada descomposición genera cuatro nuevos rectángulos idénticos. En algún punto o nivel de resolución, solamente las celdas cuyos interiores estén completamente en C_f son empleadas para construir el grafo de conectividad. Si la búsqueda en este grafo termina satisfactoriamente, se

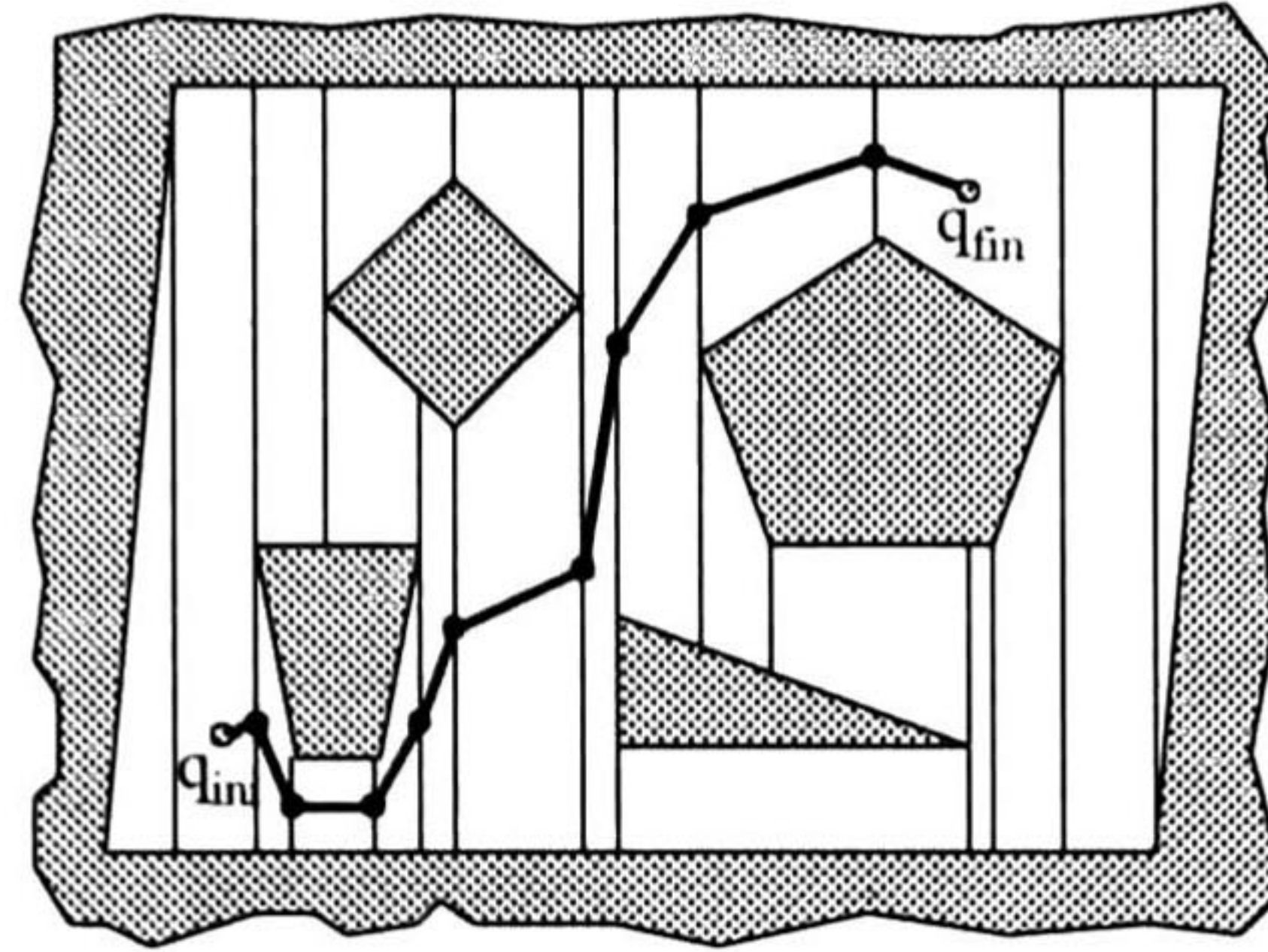


Figura 2.9: La trayectoria se puede formar con las configuraciones de los puntos medios de los segmentos que limitan las celdas, como lo muestra esta figura.

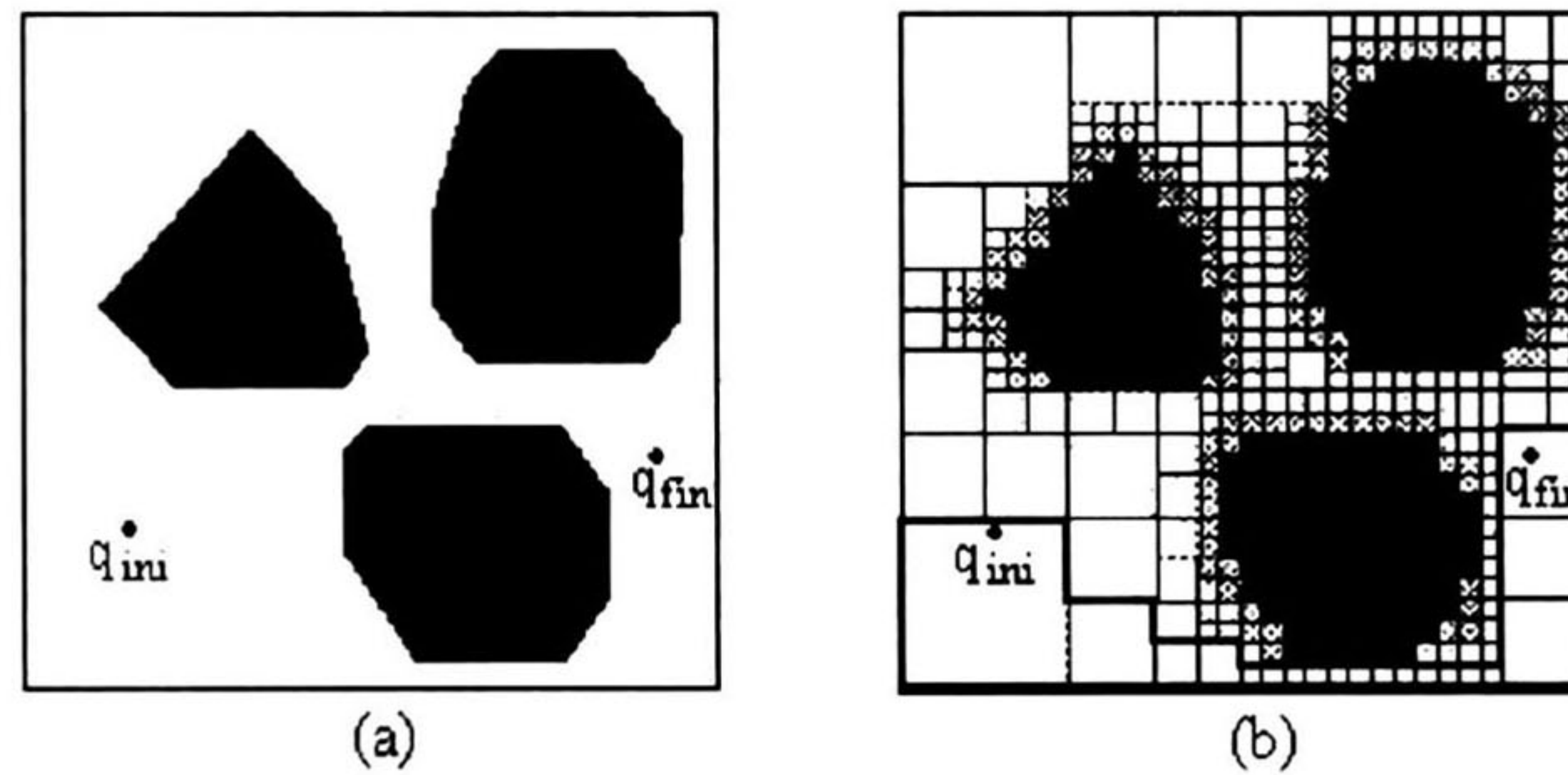


Figura 2.10: Descomposición aproximada de celdas.

genera fácilmente una trayectoria en C_f . De lo contrario, significa que la resolución de la descomposición no fue suficiente, o que no existe una trayectoria entre las configuraciones inicial y final. Frecuentemente algunos métodos de descomposición aproximada de celdas operan de una manera jerárquica empleando una descomposición no fina al principio, y refinándola poco a poco hasta que se encuentre una trayectoria o hasta llegar a un límite de resolución.

Considerando que los métodos anteriores de descomposición de celdas cuentan con técnicas apropiadas de búsqueda y de cálculos de la trayectoria, los métodos de descomposición exacta de celdas son completos, es decir, garantizan encontrar una trayectoria siempre que esta exista o indicar una falla si no existe. Los métodos aproximados pueden no ser completos; pero para la mayoría de ellos, la precisión de la aproximación se puede ajustar y hacer arbitrariamente pequeña, con la desventaja de que se consume mayor tiempo, estos métodos se dice que son de *resolución completa*. Por otro lado, los métodos exactos son matemáticamente más complejos que los aproximados, así que estos últimos son más sencillos de implementar.

CAMPO POTENCIAL.

Una aproximación más directa para la planeación del movimiento es discretizar el espacio de configuraciones en una malla regular y fina de configuraciones, y buscar en esta malla una trayectoria. Este método requiere técnicas muy poderosas de heurística para guiar la búsqueda, ya que la malla de configuraciones es en general enorme. Se han propuesto diferentes técnicas heurísticas, entre las cuales las más importantes toman la forma de funciones que son interpretadas como *campos potenciales*.

La interpretación de la terminología es que el robot, representado como un punto en el espacio de configuraciones, es una partícula moviéndose bajo la influencia de un campo potencial artificial producido por la configuración final y los C-obstáculos. En general, la configuración final genera un “potencial atractivo” que atrae al robot hacia la configuración final, y los C-obstáculos producen un “potencial repulsivo” que repelen al robot de ellos. El gradiente negado del potencial total es aplicado como una fuerza artificial aplicada al robot. En cada configuración, la dirección de esta fuerza es considerada la dirección más indicada del movimiento.

La figura 2.11 muestra la noción de potenciales atractivos y repulsivos, y su combinación. El potencial atractivo (fig. 2.11.b) es un plano cóncavo con su mínimo en la configuración final. El potencial repulsivo (fig. 2.11.c) es no cero solamente cuando se está a cierta distancia de los obstáculos y tiende a infinito cuando la distancia a C-obstáculo tiende a cero. Una trayectoria entre la configuración inicial y final se construye siguiendo el gradiente negado del potencial total (fig. 2.11.e). Una matriz de orientaciones del campo vector del gradiente negado (orientaciones de las fuerzas artificiales inducidas por el campo potencial) se muestra en la figura 2.11.f.

En comparación con otros métodos, los métodos de campo potencial pueden ser muy eficientes. Sin embargo, requieren de un mayor número de consideraciones. Debido a que son métodos rápidos con una cierta optimización, pueden quedar “atrapados” en mínimos locales de las funciones de potencial y no llegar a la configuración final. Una manera de resolver este problema es diseñar funciones de potencial sin mínimos locales que no sean la configuración final. Otra manera es complementar la aproximación de campo potencial básica con poderosos mecanismos de escape de los mínimos locales. Thorpe (1984a; 1984b) emplea una búsqueda basada en rejillas (grid-based search) para encontrar una trayectoria de costo mínimo hacia la meta, ajustando la trayectoria atrapada hacia la trayectoria de costo mínimo, después ejecuta la trayectoria con una variante de campo potencial para mantener el vehículo en la trayectoria alejado de la “trampa”. Krogh and Thorpe (1986) proponen la integración de un esquema de evasión de colisiones de campo potencial generalizado con un planeador de trayectorias global basado en rejillas de certeza (certainty grids) para la planeación óptima de la ruta y recuperación de “trampas”

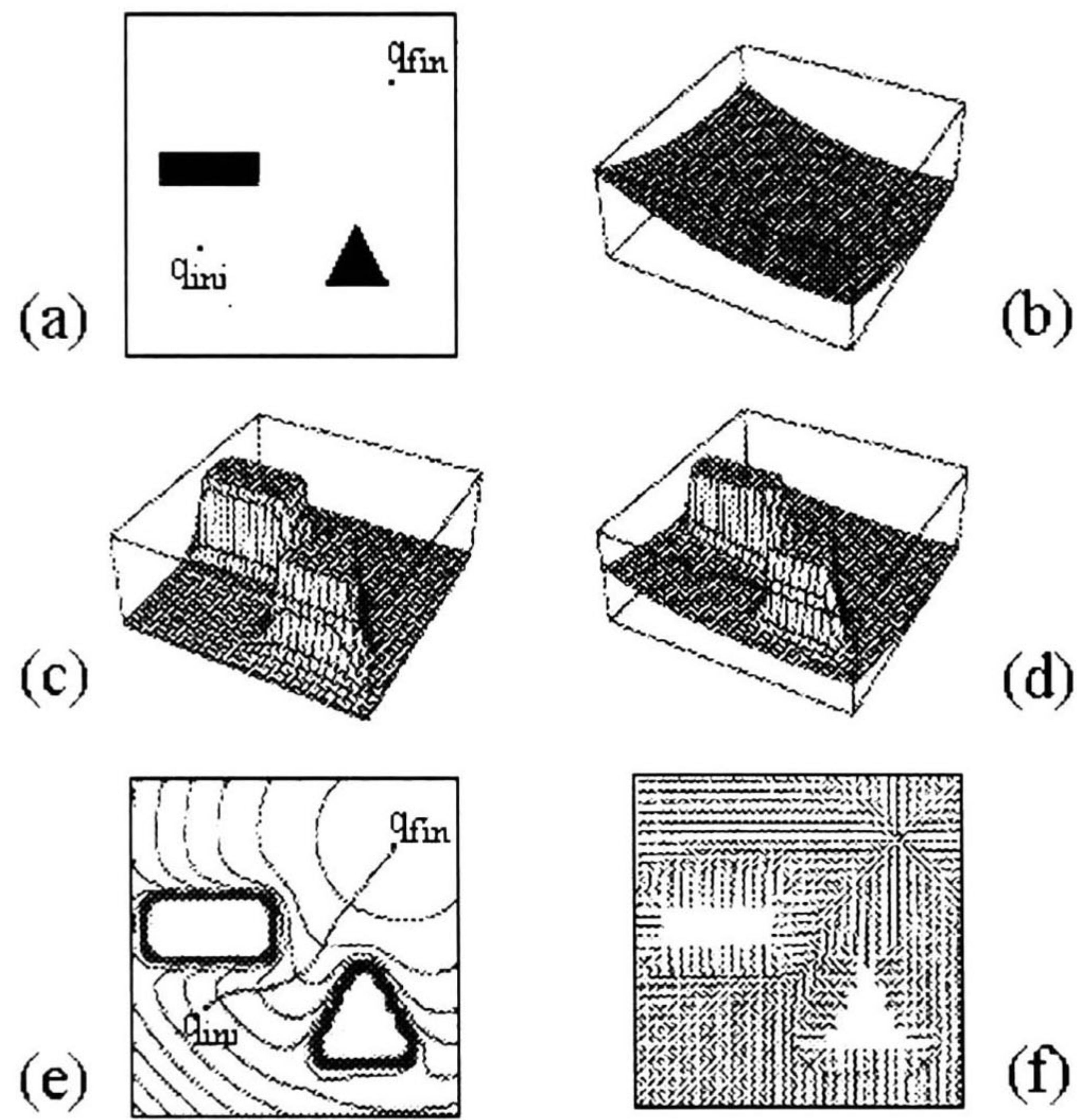


Figura 2.11: Método de campos potenciales en un espacio de configuración bi-dimensional.

2.2.3 SELECCIÓN DE LA RUTA ÓPTIMA.

Una vez seleccionada una forma de representar el conocimiento del mapa y elegir, en gran parte, la planeación del movimiento del AGV, ahora se debe seleccionar una ruta que seguirá el vehículo para resolver el problema básico. Como se mencionó en la parte anterior, la solución escogida para los problemas de representación del conocimiento del mapa involucran directamente los métodos de búsqueda de la trayectoria óptima. Se mencionó además, que el método escogido en esta tesis es un método de descomposición exacta de celdas. Este método, al igual que algunos otros lleva intrínseco el tipo de búsqueda que se realizará. Esta búsqueda se refiere precisamente a que una vez que se tiene el grafo de conectividad que representa el espacio libre del AGV, se tiene que buscar una trayectoria en el grafo que nos lleve de la configuración inicial a la configuración final. La trayectoria que se busca en el grafo es lo que llamamos anteriormente la ruta, que definimos como una secuencia de puntos en C_f , y que corresponde a una secuencia de nodos en el grafo de conectividad y que en la teoría de grafos se denomina trayectoria. Para evitar confusiones nosotros usaremos el término de ruta para definir la secuencia de nodos buscada en el grafo de conectividad que nos lleve de la configuración inicial a la configuración final.

El tipo de búsquedas que nos permiten resolver el problema de encontrar la ruta en el grafo de conectividad cae en el campo de estudio de la Inteligencia Artificial (IA) y la definición de esta área [14], es la siguiente:

Definición 25 *La Inteligencia Artificial se puede definir como la rama de la ciencia computacional que se encarga de la automatización de un comportamiento inteligente. La Inteligencia Artificial incluye la colección de problemas y metodologías empleadas por los investigadores de la IA.*

La IA ha estado muy ligada a los problemas de planeación de los robots y se han empleado muchas de sus metodologías en la solución de problemas relacionados con la planeación de los movimientos de los robots en su espacio de trabajo. Algunas de las metodologías de la IA empleadas en el área de la robótica son las búsquedas. Una herramienta usada para representar los problemas y hacer una búsqueda de la solución son los grafos, ya que los grafos pueden representar el espacio de estados de un problema [14]:

“En la representación del espacio de estados de un problema, los nodos de un grafo corresponden a estados de soluciones parciales de problemas y los arcos corresponden a los pasos o acciones en el proceso de solución de un problema. Un estado inicial, que corresponde a la información que se tiene al inicio del problema, forma el nodo inicial del grafo. El grafo también define una condición final, que es la solución al problema. La búsqueda en el espacio de estados caracteriza la solución del problema como un proceso de búsqueda de la trayectoria solución en el grafo que nos conduce del estado inicial al estado final”

En el problema de encontrar la ruta óptima, la búsqueda termina cuando se encuentra la trayectoria más corta en distancia entre los nodos inicial y final del grafo de conectividad.

Gran parte del trabajo en las búsquedas consiste en determinar un método adecuado al problema de búsqueda, y que sea apropiado en cuanto a la información que se tiene y al tipo de solución que se busca. Las estrategias de búsqueda se evalúan bajo ciertos criterios. Estos criterios se refieren generalmente a la *complejidad* del algoritmo de búsqueda, si el algoritmo es *completo*, y si el algoritmo es *óptimo*. El primer criterio se refiere al tiempo que toma encontrar la solución y a cuánta memoria se necesita para realizar la búsqueda. La completitud del algoritmo se refiere a que la estrategia de búsqueda sea capaz de garantizar que encuentra la solución cuando esta existe; en este caso se dice que el algoritmo es *completo*. La optimalidad de la estrategia de búsqueda garantiza que la solución encontrada es la más adecuada o la de mayor calidad, bajo cierto criterio de optimización, cuando existen diferentes soluciones.

Existen dos grupos fundamentales en que se dividen las estrategias de búsqueda:

- a) *Búsqueda No-Informada*. Este tipo de búsqueda se refiere a que no se tiene información sobre el número de pasos o del costo de la trayectoria de un estado dado al estado final. Todo lo que puede hacer en este tipo de búsquedas es distinguir el estado final o meta de los demás estados. Este tipo de búsquedas se llaman también *búsquedas ciegas*.

Los principales métodos de búsqueda ciega son [19]:

1. *Búsqueda de Primera Expansión* (“*Breadth-First search*”).
2. *Búsqueda de Costo Uniforme* (“*Uniform-Cost search*”).
3. *Búsqueda de Primera Profundidad* (“*Depth-First search*”).
4. *Búsqueda de Profundidad Limitada* (“*Depth-Limited search*”).
5. *Búsqueda de Profundidad Iterativa* (“*Iterative Deepening search*”).
6. *Búsqueda Bidireccional* (“*Bidirectional search*”).

Sus características principales se resumen en la tabla 2.12. En esta tabla se hace una comparación de cuatro criterios de evaluación para los métodos de búsqueda no informada. Para la interpretación de la tabla comparativa entre las estrategias de búsqueda se deben dar las definiciones siguientes:

Definición 26 *Se dice que un nodo se expande cuando se avanza en el grafo y se identifican todos los nodos adyacentes siguientes a ese nodo. Los nodos adyacentes se denominan hijos y el nodo expandido se denomina padre.*

Definición 27 *La profundidad en una búsqueda sobre un grafo, es el número de arcos recorridos en una trayectoria desde el nodo raíz. Se emplea el término de profundidad para indicar cuántos nodos se han expandido en una trayectoria desde la raíz hasta un nodo, durante la búsqueda.*

CRITERIOS	Primera Expansión	Costo Uniforme	Primera Profundidad	Profundidad Limitada	Profundidad Iterativa	Bidireccional
Tiempo	b^d	b^d	b^m	b^l	b^d	$b^{d/2}$
Espacio	b^d	b^d	bm	bl	bd	$b^{d/2}$
Óptimo	Sí	Sí	No	No	Sí	Sí
Completo	Sí	Sí	No	Sí, si $l \geq d$	Sí	Sí

Figura 2.12: Comparación de estrategias de búsqueda. El factor de ramificación es b ; d es la profundidad de solución; m es la profundidad máxima de búsqueda; l es el límite de profundidad de búsqueda.

Definición 28 El factor de ramificación (branching factor) es el número de nodos nuevos que se obtienen cada vez que se expande un nodo en un grafo.

- b) *Búsqueda Informada*. En este tipo de búsquedas el agente² tiene más información, como puede ser el costo aproximado para ir de un nodo cualquiera al nodo final, de esta manera el agente puede hacer una elección más inteligente del siguiente paso que debe escoger. Este tipo de estrategias también se llaman *búsquedas heurísticas*. Una cuestión importante en estas búsquedas es saber en dónde introducir la información del conocimiento que se tiene para aprovechar esta información en la búsqueda. La alternativa es la implementación de funciones que ayuden a determinar la mejor opción del siguiente nodo. De esta manera, el conocimiento que hace esta determinación es proporcionado por una *función de evaluación* que entrega un número proporcional que representa la ventaja o desventaja de expandir un nodo. En esta metodología se incluyen las llamadas *funciones heurísticas* que estiman costos aproximados desde un nodo cualquiera al nodo meta, con esta información extra se puede evaluar un nodo que esté más cercano a la meta y que sea el mejor prospecto a ser expandido. La heurística se ha considerado importante desde los inicios de las aplicaciones de la IA. El primer *sistema experto*³ de laboratorio, iniciado en la Universidad de Stanford por Ed Feigenbaum, Bruce Buchanan y Joshua Lederberg, fue llamado *Heuristic Programming Project (HPP)*. La heurística en el campo de la IA ha sido visualizada como “reglas de la experiencia” que los sistemas expertos pueden usar para generar soluciones aceptables sin exhaustivas búsquedas. Actualmente, el término heurística se emplea como adjetivo, refiriéndose a cualquier técnica que mejore el desempeño de los casos promedio en

²Un agente, en el campo de la IA, se puede definir [Russell & Norving, 1995] como cualquier elemento que pueda percibir su entorno a través de sensores y actuar en ese entorno por medio de efectores. Con esta definición podemos considerar a un AGV como un agente.

³“Son sistemas o programas que ayudan a la toma de decisiones y que en principio contienen el conocimiento, o parte del conocimiento, de un experto” [Benchimol & Levine, 1990]

una tarea de resolver un problema, pero no necesariamente mejorando el peor de los casos. En el área específica de algoritmos de búsqueda, se refiere a una función que proporciona un estimado del costo de una solución, pero no necesariamente el costo exacto.

En el simulador de esta tesis, empleamos un método de búsqueda informada en el cual la información que tenemos es la distancia entre los nodos adyacentes, que se representan como los pesos en los arcos del grafo de conectividad. Además se emplea una función heurística que estima la distancia desde cada uno de los nodos a la configuración final, para ayudar a decidir el siguiente nodo expandido durante la búsqueda de la ruta óptima del AGV. En el siguiente Capítulo se describe detalladamente el algoritmo empleado y sus características.

2.2.4 DETERMINACIÓN DE LA TRAYECTORIA.

Cuando se ha determinado la ruta que tomará el AGV generalmente surge el problema de establecer la trayectoria que se va a seguir. Con la ruta lo que se tiene es una secuencia de puntos en el espacio libre del AGV, sin embargo estos puntos no corresponden de manera específica a configuraciones en la trayectoria del AGV. Esto no es una regla y, como se ha mencionado con anterioridad, esto depende de la forma de resolver el problema de determinar la trayectoria que seguirá el vehículo y de la forma en que se diseñe un generador de trayectorias a partir de una secuencia de nodos en el grafo de conectividad.

La idea básica en esta etapa es que, partiendo de la información que nos da un algoritmo de búsqueda en el grafo de conectividad, se establece una ruta óptima que debe ser cubierta por el vehículo para llegar de la configuración inicial a la configuración final. Sin embargo, la secuencia de nodos que establece la ruta óptima en muchos casos no es adecuada para ser la trayectoria final que recorrerá el vehículo. En la mayoría de las ocasiones, se prefiere modificar la trayectoria del grafo de conectividad para suavizar las curvas y obtener una trayectoria mucho más apropiada para un vehículo. La razón por la que se desea suavizar la trayectorias de los grafos es evidente si consideramos que los grafos establecen trayectorias formadas por nodos y arcos que son segmentos de líneas rectas y algunas curvas que provocarían, en muchas ocasiones, movimientos muy complicados para los vehículos por cerrados y poco prácticos. Por lo tanto, se pretende tomar como referencia la trayectoria especificada por el grafo y suavizar las curvas para obtener una trayectoria adecuada para el vehículo. (Ver la figura 2.13).

Las técnicas o métodos empleados para suavizar las trayectorias son puramente matemáticos. Los métodos empleados para determinar la trayectoria suavizada toman la información de los nodos de la ruta como referencia, con estos puntos se hace una aproximación a curvas paramétricas continuas que establecen una trayectoria en función del tiempo. Estas funciones son polinomiales y existen aproximaciones que emplean diferentes grados en los

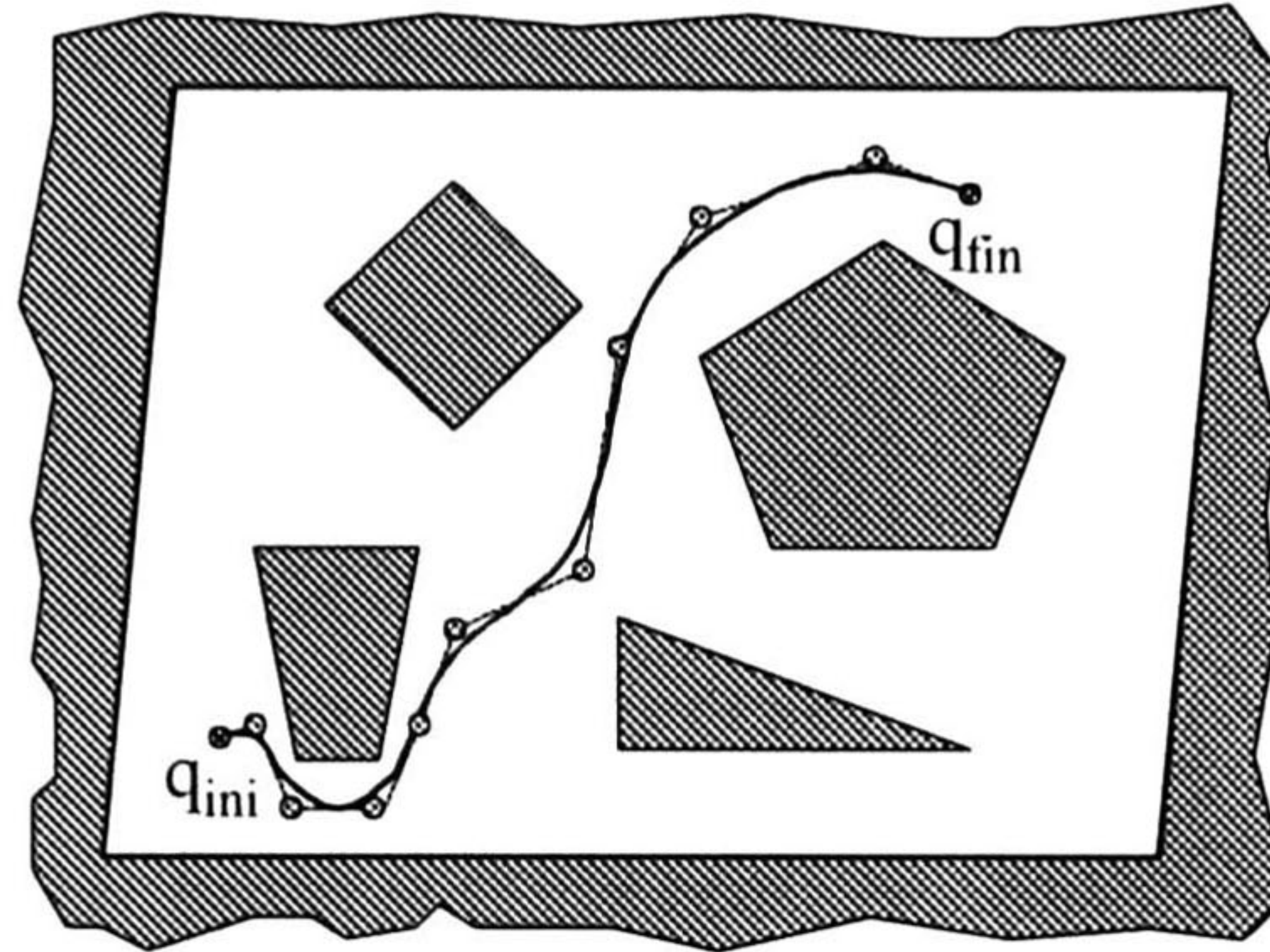


Figura 2.13: Diferencia entre la ruta y la trayectoria suavizada para el seguimiento por parte del AGV.

polinomios que se calculan. En el Capítulo 4 se describe el procedimiento de las curvas llamadas B-splines, que es el empleado en el simulador de esta tesis para suavizar la trayectoria especificada por el grafo de conectividad y obtener la trayectoria que seguirá el AGV.

2.2.5 SEGUIMIENTO DE LA TRAYECTORIA.

Cuando se tiene una trayectoria definida para el vehículo se debe hacer que, por medio de algún sistema de control, el vehículo pueda realizar la trayectoria que se desea. Este es el problema de seguimiento de la trayectoria. La trayectoria generalmente es especificada como una función del tiempo, de manera que el vehículo debe seguir esta función durante su recorrido. Para lograr un seguimiento de una trayectoria los sistemas, en general, emplean complicados algoritmos de control que en la mayoría de las ocasiones incluyen un modelo matemático del propio sistema. Existen diferentes métodos que dependen de la dinámica del sistema que se quiere controlar y de la complejidad del sistema. El estudio del problema de seguimiento de trayectorias de sistemas es un campo muy extenso de estudio de la Teoría de Control de Sistemas. Lo que se pretende en estos esquemas es controlar las salidas del sistema para que sigan las referencias deseadas, que se denominan trayectoria del sistema. Para esto se introduce alguna ley de control que toma las salidas del sistema y entrega como salida las entradas a los actuadores del sistema para que siga la trayectoria deseada.

En un AGV inteligente las trayectorias que sigue el sistema son las trayectorias que seguirá el vehículo en su área de trabajo, mientras que el sistema que se quiere controlar es el propio vehículo y la salida del sistema será la configuración del vehículo en el espacio libre. En el simulador de esta tesis se emplea un modelo matemático del AGV para poder calcular

las nuevas condiciones de entrada del sistema. El esquema implementado en el seguimiento de la trayectoria es descrito en el Capítulo 4.

2.2.6 TAREAS REACTIVAS.

El nombre de tareas reactivas se refiere a aquellas funciones que realizan ciertos tipos de AGV al interactuar con su medio ambiente y presentarse situaciones no previstas durante la etapa de planeación del movimiento del vehículo. Estas funciones son las tareas de reacción que desempeñará el vehículo ante situaciones tales como: batería baja, detección de algún obstáculo durante su recorrido, problemas de señal en los cables guía, etc. En estas tareas deben ser incluidas pues, todas aquellas posibles situaciones que no se contemplan durante la planeación del movimiento del vehículo y que pueden provocar que el vehículo no cumpla con su recorrido satisfactoriamente, poniendo en riesgo la seguridad de las personas, de la carga o del propio vehículo.

La solución de las tareas reactivas de un AGV dependen del tipo de vehículo y su aplicación particular, ya que estas características sirven para definir las funciones que realizará el vehículo ante situaciones que pongan en riesgo la consecución de la meta del problema básico. Por ejemplo, los AGV para cargas pesadas con cable guía enterrado en el piso, por cuestiones de seguridad, al detectar algún obstáculo durante su recorrido se detienen y activan algún tipo de alarma visual y auditiva. Por el contrario, algunos vehículos de cargas ligera también con cable guía son capaces de realizar algunas rutinas de evasión de obstáculos. En el AGV simulado, consideramos dentro de las tareas reactivas del vehículo la evasión de obstáculos que son detectados mediante sensores de ultrasonido también simulados en el programa. En las tareas reactivas, el problema de la evasión de obstáculos es el que concentra la mayor atención en el desarrollo de algoritmos, ya que las demás tareas reactivas comprenden fundamentalmente la activación de alarmas para que el personal técnico atienda la situación detectada. Los obstáculos móviles, que no son considerados por el algoritmo de planeación de la trayectoria del vehículo, se pueden evadir empleando tareas reactivas que trabajan en paralelo con el algoritmo generador de la trayectoria al detectar estos obstáculos, esto conduce a dividir el proceso de planeación del movimiento en una planeación de la trayectoria y una detección de obstáculos durante el recorrido de tal trayectoria.

Podemos hacer una división entre los obstáculos [19] :

- *Obstáculos Estáticos*: Son aquellos que están fijos en todo tiempo.
- *Obstáculos Parcialmente Dinámicos*: Son obstáculos que se encuentran en posiciones constantes solo por ciertos periodos de tiempo.

- *Obstáculos Completamente Dinámicos*: Son aquellos obstáculos que se están moviendo alrededor del espacio de trabajo del vehículo en direcciones desconocidas.

En los algoritmos previos al movimiento del AGV, se pueden considerar sólo los obstáculos estáticos. La incorporación de la posición actual de los obstáculos dinámicos no tiene sentido en esta etapa, ya que estos obstáculos podrían estar en otra posición durante el recorrido de la trayectoria planeada. Por el contrario, durante el recorrido, el AGV debe tomar en consideración todos aquellos obstáculos que se presenten en su entorno local, especialmente los obstáculos completamente dinámicos, para modificar la trayectoria de ser necesario. El problema de evasión de obstáculos es crítico y muy complejo, ya que existen muchos tipos de colisiones potenciales durante el recorrido de una trayectoria que se deben resolver. Es importante tener en cuenta que los algoritmos de generación de trayectorias durante el recorrido pueden llegar a modificar la trayectoria de manera que ésta, finalmente no coincida con la trayectoria óptima determinada en la solución de los algoritmos de planeación previos. El Capítulo 5 trata específicamente de las tareas reactivas del simulador y su implementación en la evasión de obstáculos.

Capítulo 3

REPRESENTACIÓN DEL MAPA Y BÚSQUEDA DE LA RUTA ÓPTIMA.

En este capítulo se verán los temas de representación del mapa y búsqueda de la ruta óptima que constituyen la etapa de preplaneación del movimiento del vehículo. El término *preplaneación del movimiento* se refiere a la etapa del algoritmo de control de navegación que se ejecuta antes de iniciar el movimiento del vehículo; mientras que a los algoritmos que actúan durante el movimiento son los que constituyen la etapa que denominaremos *planeación del movimiento*. En la primera sección se tratará la manera en que se representa la información del mapa para el simulador desarrollado. En el capítulo anterior se presentaron varios métodos y se mencionó que empleamos la descomposición exacta de celdas para representar el espacio libre del vehículo mediante un grafo de conectividad que nos representa la relación de adyacencia entre las celdas formadas y nos proporciona un mecanismo adecuado para realizar en él una búsqueda de la ruta óptima. La segunda parte de este capítulo está dedicada a la explicación del algoritmo empleado para realizar la búsqueda de la ruta óptima en el grafo de conectividad generado. Como se mencionó anteriormente se emplea un algoritmo de búsqueda de los denominados informados, que emplea una función heurística como parte de su función de evaluación para determinar la ruta óptima. Esta ruta que buscamos en el grafo de conectividad es la ruta óptima en distancia, es decir la ruta más corta para que el vehículo llegue de la configuración inicial a la configuración final. Debido a esto, el peso de los arcos del grafo de conectividad es la distancia entre los nodos adyacentes. En la etapa de planeación del movimiento de los AGV, la manera en que se representa el mapa y se establece la ruta a seguir es la parte que determina el funcionamiento del vehículo y marca en gran medida el resto del trabajo de planeación. Los algoritmos posteriores a la preplaneación dependen de la elección de los métodos de representación de la información del mapa y la búsqueda empleada, ya que se deben implementar aquellos algoritmos que puedan compartir el mismo tipo de información para generar la trayectoria y realizar las tareas reactivas del vehículo. Estos algoritmos posteriores a la generación de la ruta óptima,

se utilizan durante el recorrido y se explican en los siguientes capítulos.

3.1 REPRESENTACIÓN DEL MAPA.

La representación del mapa del espacio de trabajo de un AGV inteligente es la parte más importante en la preplaneación del movimiento. Su importancia radica en el hecho de que es la parte inicial de todo un proceso de planeación y control de movimientos del vehículo que involucra varios algoritmos; esto hace que la información básica de todo el proceso dependan de esta representación de la información del mapa. Todos los algoritmos posteriores a éste deben adecuarse y ser capaces de emplear esta forma de representación de la información del mapa para realizar sus respectivas tareas. Si por ejemplo, se decide implementar un método de campos potenciales, como el mencionado en el capítulo anterior, los algoritmos necesarios para la generación de la trayectoria del vehículo para este método son muy diferentes a los empleados en un caso de descomposición exacta de celdas, el cual es el empleado en esta tesis.

3.1.1 MÉTODO DE DESCOMPOSICIÓN EXACTA DE CELDAS.

El principio del método de *descomposición exacta de celdas* es descomponer primero el espacio libre del vehículo C_f en un conjunto de regiones que no se traslapan entre sí, llamadas *celdas*, cuya unión corresponde exactamente a la región C_f . Con las celdas se forma posteriormente un grafo que representa la relación de adyacencia entre las celdas, llamado *grafo de conectividad*. En el grafo de conectividad se realiza la búsqueda de una ruta para el vehículo que conecte las configuraciones inicial y final. La estrategia de búsqueda en el grafo de conectividad se discute en la siguiente sección de este capítulo.

Las celdas generadas por la descomposición exacta del espacio libre del vehículo deben cumplir con las siguientes características [13]:

1. La geometría de cada celda debe ser tan simple que permita determinar con facilidad una trayectoria entre cualquier par de configuraciones en la celda.
2. No debe ser difícil establecer la adyacencia entre dos celdas cualesquiera y encontrar una trayectoria que cruce la región del límite que comparten las celdas adyacentes.

Estas características se pueden interpretar de manera que sugieren que los límites de las celdas correspondan a cambios “críticos” en C_f , es decir, algún cambio significativo que se presenta cuando se cruza el límite. Así mismo, las celdas serán regiones “no-críticas”,

de tal manera que todas las trayectorias dentro de una misma celda para cualquier par de configuraciones dentro de ella son fácilmente realizables con trayectorias rectas.

Existen diferentes métodos de descomposición exacta de celdas, las principales diferencias radican en la manera de considerar el espacio libre de los robots. Existen por ejemplo, métodos que consideran regiones C-obstáculo tridimensionales; otros resultan complicados al establecer modelos matemáticos para limitar las regiones C-obstáculo por reglas de superficies. En realidad algunos métodos son inapropiados para la implementación y como lo mencionan algunos autores como Schuartz y Sharir [13]: “Algunos métodos resultan catastróficamente ineficientes, y principalmente sirven para probar la existencia de los métodos e ilustrar la aplicación de los mecanismos de decisión” El método implementado en esta tesis es un método aplicable a problemas donde la configuración del espacio de trabajo es bi-dimensional y los C-obstáculos son poligonales. Si un obstáculo presenta alguna curvatura (no es poligonal), la sección curva se puede aproximar con un segmento poligonal de manera que la curvatura quede inscrita en la región poligonal trazada. Este método permite hacer una descomposición sencilla de celdas simples trapezoidales. El grafo de conectividad que representa la relación de adyacencia entre celdas determina una red de curvas o de segmentos rectilíneos, que se puede obtener seleccionando un punto arbitrario que represente a cada celda, y uniéndolo por medio de una línea al punto que representa a una celda d con el punto que representa a cada celda adyacente a d . De esta manera los puntos que representan a las celdas forman los nodos y las líneas forman los arcos del grafo de conectividad. Esta representación proporciona al vehículo información muy útil para generar la trayectoria del vehículo y poder afrontar problemas de obstáculos dinámicos en su recorrido, esto se explicará más adelante en los capítulos 4 y 5.

La descomposición del espacio C_f y el grafo de conectividad asociado se presentan formalmente en las siguientes definiciones [13]:

Definición 29 Una descomposición poligonal convexa K de C_f es un conjunto finito de polígonos convexos, llamados celdas, tal que los interiores de cualesquiera dos celdas no se intersectan y la unión de todas las celdas es igual a la cerradura de C_f , $cl(C_f)$. Dos celdas k y k' son adyacentes si y solo si $k \cap k'$ es un segmento de línea de longitud no cero.

Definición 30 El grafo de conectividad G asociado con una descomposición poligonal convexa K de C_f es el grafo no dirigido especificado como sigue:

- Los nodos de G son las celdas en K .
- Dos nodos en G son unidos por un arco si y solo si las celdas correspondientes son adyacentes.

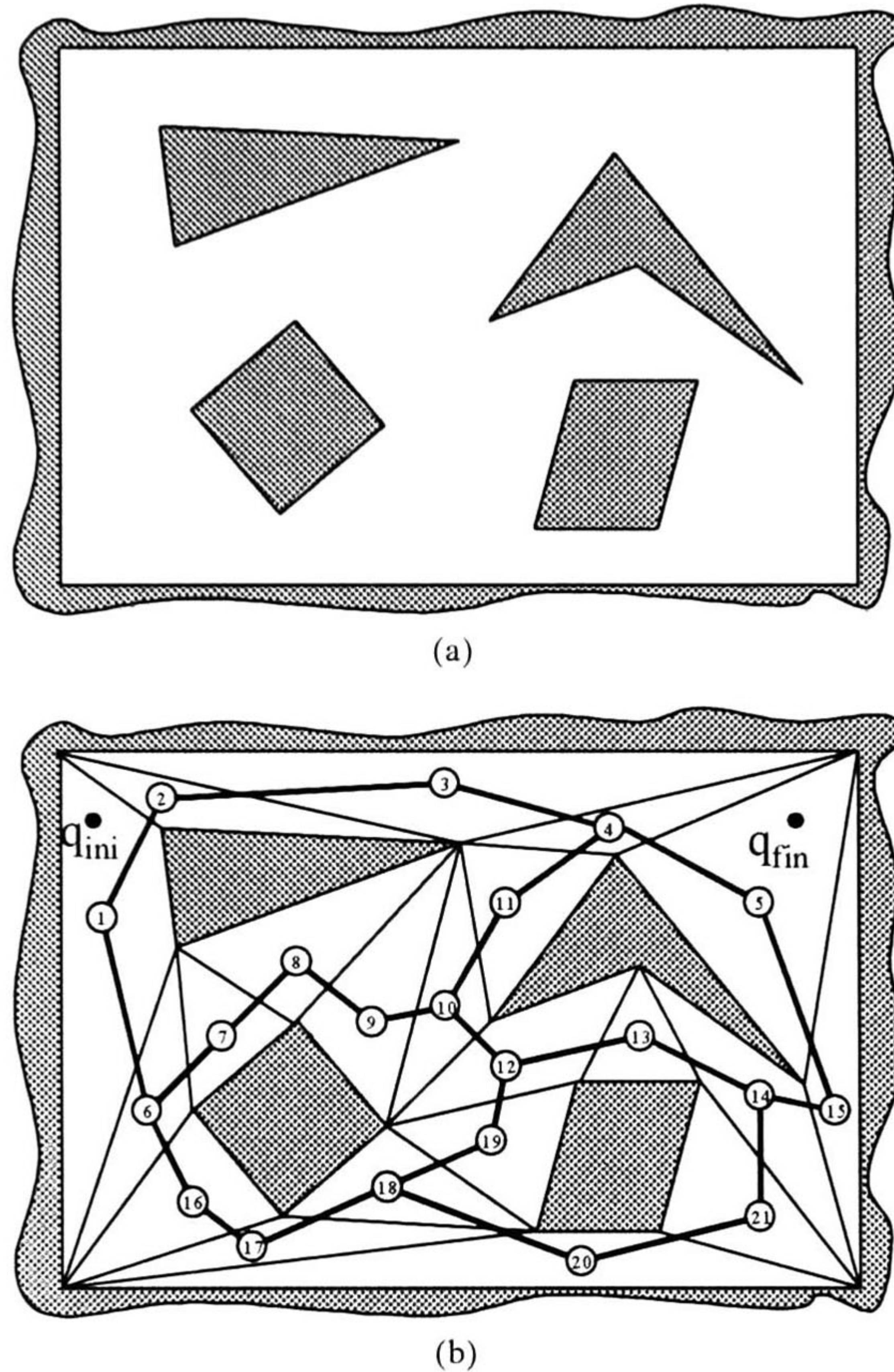


Figura 3.1: La figura (a) muestra el espacio de configuración bi-dimensional con obstáculos poligonales. La figura (b) muestra la descomposición del espacio libre C_f .

La figura 3.1 muestra una descomposición poligonal convexa. En la figura 3.1.a se muestra el espacio libre del robot y los obstáculos poligonales. La figura 3.1.b muestra la descomposición en celdas del espacio libre y el grafo de conectividad asociado.

Si consideramos una configuración inicial q_{ini} , y una configuración final q_{fin} en C_f , el algoritmo de descomposición exacta de celdas para planear una ruta que conecte las dos configuraciones es el siguiente:

1. Generar la descomposición poligonal convexa K de C_f .
2. Construir el grafo de conectividad G asociado con K .
3. Hacer una búsqueda en G para determinar una secuencia de celdas adyacentes entre q_{ini} y q_{fin} .
4. Si la búsqueda termina satisfactoriamente, dar la secuencia generada de celdas como la ruta a seguir; de lo contrario, regresar "falla"

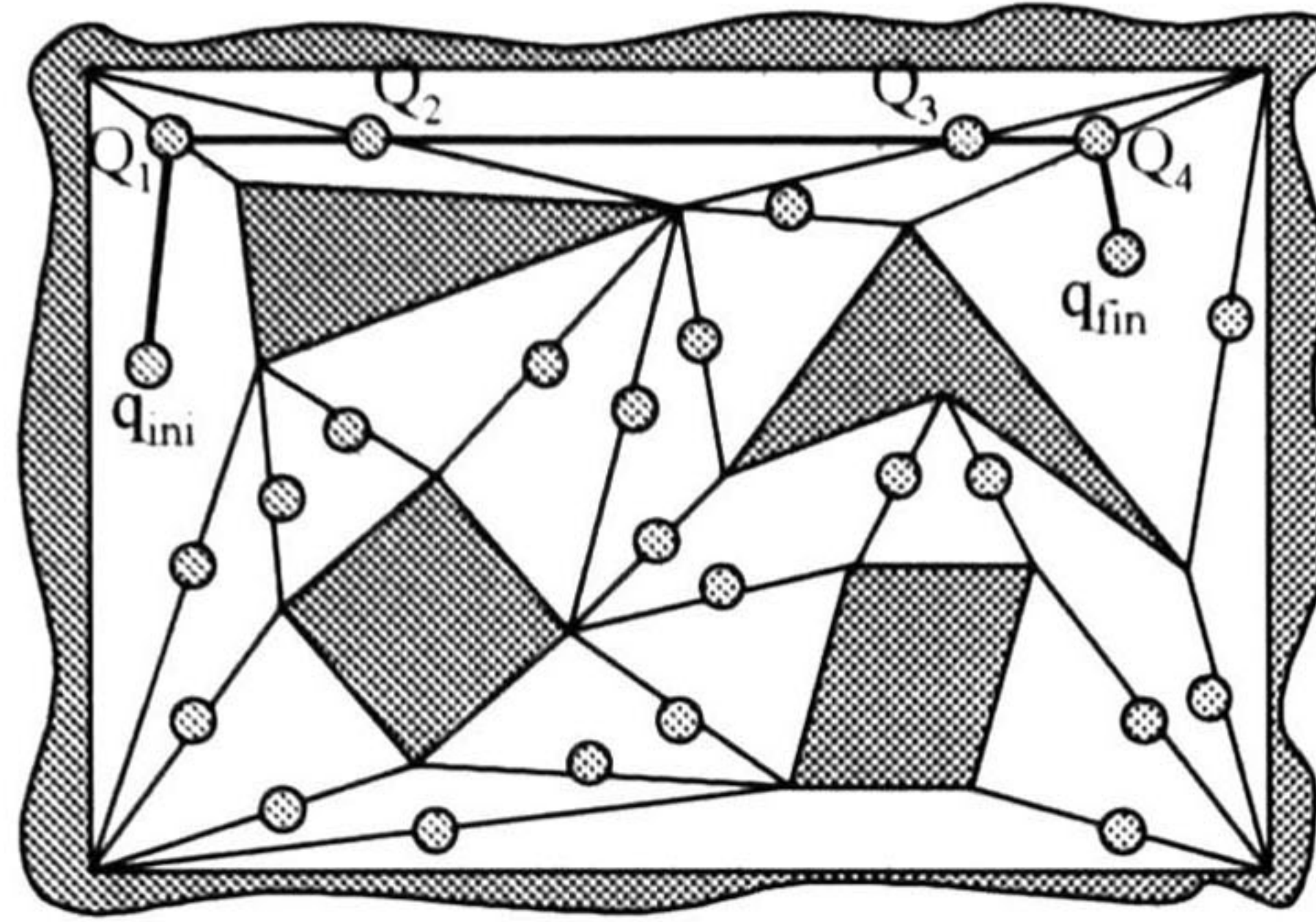


Figura 3.2: Trayectoria simple.

La salida o solución del algoritmo es una secuencia k_1, \dots, k_p de celdas tal que $q_{ini} \in k_1$, $q_{fin} \in k_p$ y para cada $j \in [1, p-1]$, k_j y k_{j+1} son adyacentes. Esta secuencia es llamada *canal*. En la figura 3.1, q_{ini} está en la celda 1 y q_{fin} está en la celda 5; un canal para este caso es la secuencia de celdas 1, 2, 3, 4 y 5.

El interior de un canal está siempre en el espacio libre. Sea β_j el segmento de línea no cero que forma la intersección entre las celdas k_j y k_{j+1} . Una manera simple de formar una trayectoria contenida en el interior de un canal producido por la búsqueda en G es considerar los puntos medios Q_j de cada segmento β_j , y conectar las configuraciones inicial y final por una línea poligonal cuyos vértices sucesivos sean los puntos Q_1, \dots, Q_{p-1} . Esta manera de construir la trayectoria se muestra en la figura 3.2. Posteriormente la trayectoria se puede suavizar por técnicas de splines, de lo cual trata el capítulo siguiente.

Ahora bien, bajo este principio básico existen varias formas de generar las celdas y el grafo de conectividad. Lo que se ha mencionado hasta ahora no indica la manera de formar las celdas y lo que se ha dicho sobre el grafo de conectividad es que debe haber un nodo por cada celda y que dos nodos son unidos por un arco si estos representan a dos celdas adyacentes. En la sección siguiente se establece la manera de generar las celdas y posteriormente se explicará la generación del grafo de conectividad.

3.1.2 GENERACIÓN DE CELDAS.

La descomposición del espacio C_f en celdas no tiene una metodología normalizada definida; por el contrario, la forma en que se define la descomposición de celdas propicia diferentes métodos para hacer una descomposición de celdas del espacio libre. Por ejemplo, los algoritmos que buscan una descomposición óptima de C_f constituyen un problema NP¹ Para un

¹El término NP (no polinomial), se refiere a aquella clase de problemas que tal vez no se puedan resolver en un tiempo menor a un tiempo exponencial, [Luger & Stubblefield, 1993].

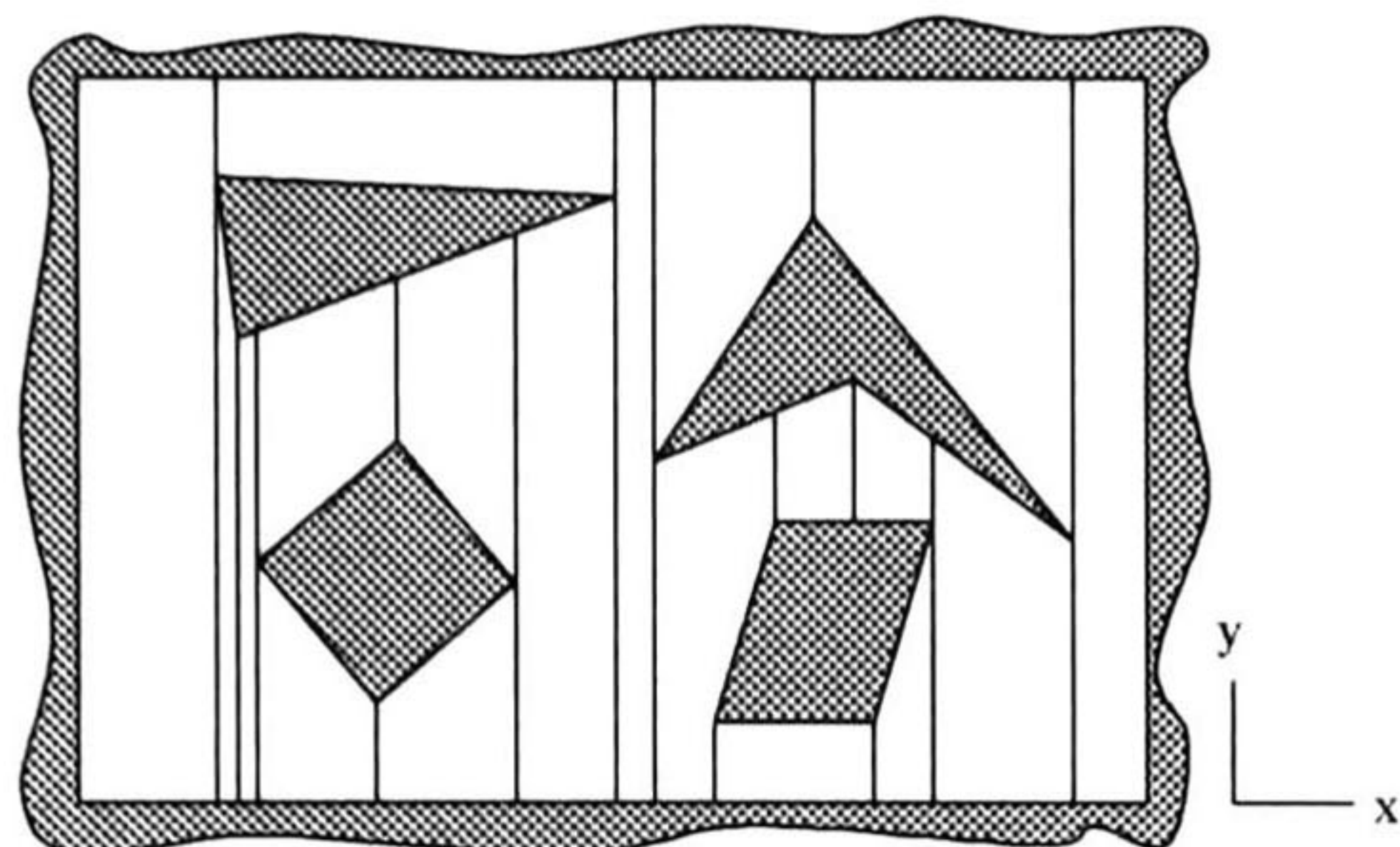


Figura 3.3: Descomposición trapezoidal del espacio C_f .

gran número de aplicaciones una descomposición exacta de celdas no óptima de C_f es más que satisfactoria.

Una descomposición fácil de implementar es la *descomposición trapezoidal*. Esta consiste en descomponer C_f trazando una línea vertical L paralela al eje “ y ” que cruza a C_f . El trazado de la línea inicia en cada vértice X de CB . La línea L es trazada a partir de X hacia arriba y hacia abajo hasta que llega a interceptar CB , así que un máximo de dos segmentos de línea vertical son trazados desde X hacia los contornos de CB , como se muestra en la figura 3.3.

Este método de formación de las celdas también se conoce como descomposición vertical [13, 20]. Cada celda formada en esta descomposición es o un triángulo o un trapecoide. Dos celdas son adyacentes si y solo si sus límites comparten un segmento vertical. Este algoritmo es aplicable tanto si C_f es limitado o si no lo es, en cuyo caso la descomposición incluye celdas que se extienden infinitamente en la dirección del eje “ y ”. En esta tesis los casos de descomposición de celdas se llevan a cabo siempre dentro de un espacio C_f limitado.

La construcción de celdas descrita anteriormente es en principio la implementada en el simulador de esta tesis; sin embargo durante la etapa de implementación se tomaron en cuenta algunos casos especiales en los que se presenta la necesidad de dividir una *celda exterior de doble segmento* en dos o más celdas para mejorar el algoritmo de generación del grafo de conectividad. Antes de la explicación definiremos a las celdas exteriores de doble segmento.

Definición 31 *Las celdas exteriores son las celdas más externas a la derecha e izquierda de C_f . Las celdas externas derechas se limitan a su izquierda por segmentos L y a la derecha por los límites propios de C_f . Las celdas externas izquierdas son limitadas a la derecha por segmentos L y a la izquierda por los límites de C_f .*

Definición 32 Una celda exterior de doble segmento es la celda exterior, a la derecha o izquierda, de C_f y que está limitada por los segmentos de línea verticales formados ya sea por un vértice X que contiene dos segmentos L_1 y L_2 (uno hacia arriba y otro hacia abajo), o bien por dos o más vértices $X_1, \dots, X_n (n \geq 2)$ que están alineados verticalmente.

Definición 33 Los vértices exteriores son aquellos vértices X_t cuyos segmentos L limitan a una celda exterior de doble segmento.

El método implementado como descomposición de celdas es el método trapezoidal ya descrito con la variante adicional siguiente:

Si existe una celda exterior de doble segmento debe ser dividida de acuerdo a los criterios siguientes:

- Si la celda es formada por los segmentos de un solo vértice X , L_1 y L_2 , la celda debe ser dividida en dos celdas trazando un segmento L_x horizontal, paralelo al eje "x", a partir de X hasta que llega a interceptar CB.
- Si la celda es formada por segmentos de dos o más vértices $X_1, \dots, X_n (n \geq 2)$ alineados verticalmente, la celda será dividida en $n+1$ celdas trazando n segmentos horizontales a partir de cada uno de los n vértices hasta que cada uno de estos llega a interceptar CB.

NOTA: La finalidad de los segmentos horizontales es dividir solamente la celda exterior.

La figura 3.4 muestra la descomposición de celdas exteriores de doble segmento donde la celda exterior izquierda queda dividida en tres celdas al trazar los segmentos horizontales y la celda exterior derecha quedará dividida en dos celdas al trazar el segmento horizontal punteado. Dependiendo de la forma y distribución de los C-obstáculos, un espacio libre puede o no tener celdas exteriores de doble segmento, incluso si llega a tener, pueden ser dos o una sola. La razón de la división de las celdas exteriores de doble segmento es por la manera de generar el grafo de conectividad de C_f , esto quedará más claro en la siguiente sección.

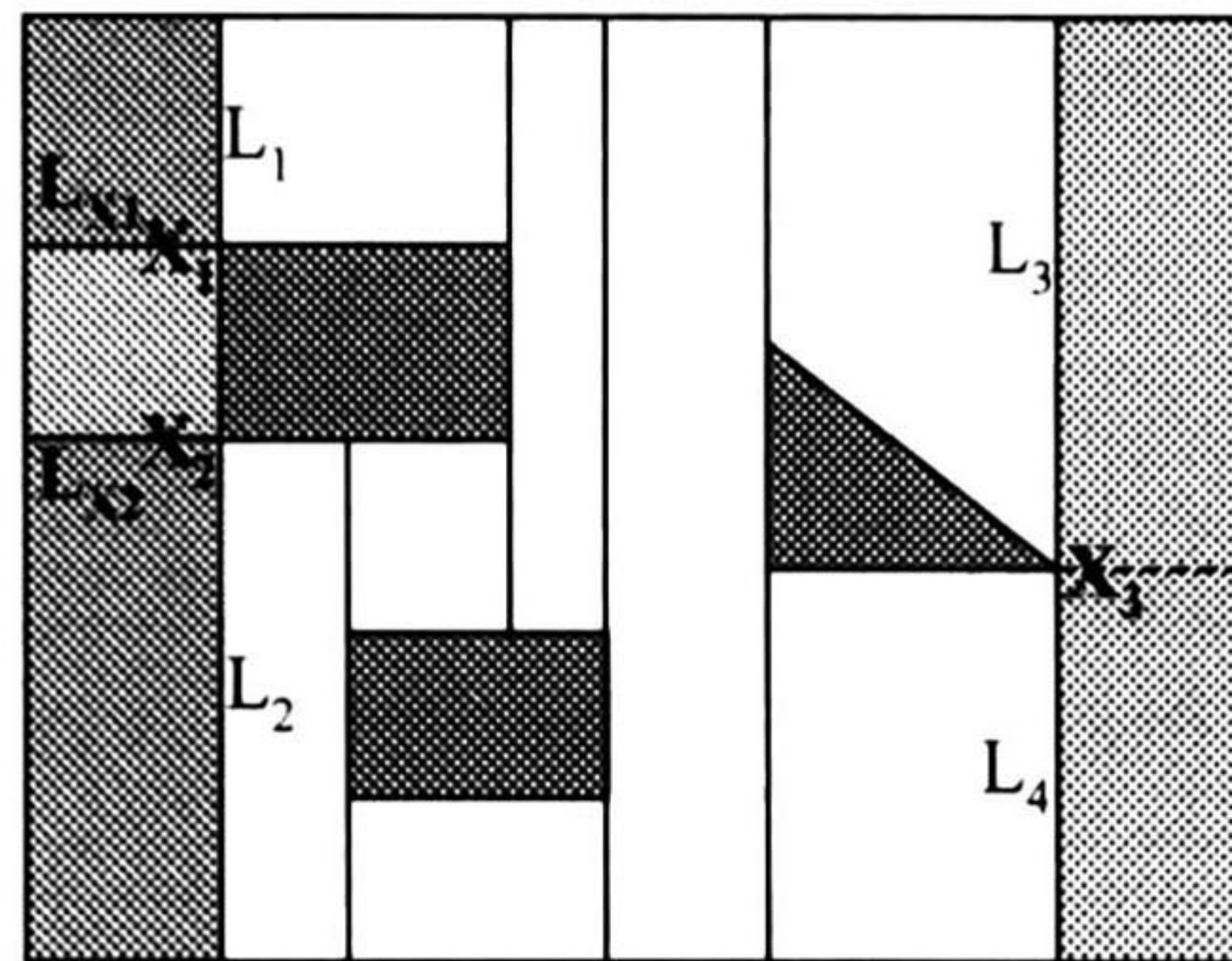


Figura 3.4: Celdas exteriores de doble segmento izquierdas y derechas.

3.1.3 GENERACIÓN DEL GRAFO DE CONECTIVIDAD.

Una vez que se forman las celdas, se debe crear el grafo de conectividad G que represente la relación de adyacencia de las celdas formadas. Este grafo resume en cierta manera la información necesaria del mapa de C_f para realizar la búsqueda de la ruta que seguirá el vehículo, para poder cruzar el espacio de configuraciones y viajar de la configuración inicial a la configuración final. La búsqueda en el grafo G puede hacerse de diferentes maneras y la sección siguiente trata del método implementado. El grafo de conectividad debe ser lo más representativo posible de C_f , así que es importante la decisión de la ubicación de los nodos del grafo en las celdas de C_f . Es conveniente que la ruta óptima determinada a partir del grafo sea lo más aproximado posible a la ruta óptima con respecto a las longitudes Euclidianas de C_f , así que el grafo de conectividad se puede especificar como sigue:

1. Los nodos de G son q_{ini} , q_{fin} , y los puntos medios Q_j de todos los segmentos L de los límites compartidos por celdas adyacentes.
2. Dos nodos son unidos por un arco no dirigido si y solo si ambos pertenecen a la misma celda.
3. Los arcos se forman por líneas rectas que unen los nodos correspondientes, y se les asigna a cada arco un peso que corresponde a la distancia Euclidiana de la línea recta que une a los nodos.

En el grafo formado de esta manera se realiza la búsqueda de la ruta entre q_{ini} y q_{fin} para posteriormente generar la trayectoria del vehículo.

Durante el proceso de implementación del algoritmo anterior se realizaron algunas modificaciones y adiciones en el método de generación del grafo de conectividad descrito anteriormente. En las especificaciones adicionales realizadas se incluyen algunas soluciones a casos específicos donde el método anterior resulta limitado, con el fin de crear un grafo más apropiado para nuestro simulador. Las especificaciones adicionales al método anterior comprenden, por ejemplo, casos en que se asigna más de un nodo a un segmento de línea L , se

omiten nodos en algunos segmentos L , se omiten arcos entre algunos nodos que pertenecen a la misma celda, se asignan *nodos de carga-descarga* a posiciones definidas en C_f , se determinan *nodos de enlace* en las entradas de *salones*, y se determina en qué salones se realiza la descomposición de celdas. La razón de incluir especificaciones adicionales al formar el grafo de conectividad es para representar de una mejor manera el espacio libre del vehículo evitando, en la mayor medida posible, rutas irrealizables que pueden ser omitidas en la etapa de preplaneación. Primeramente se darán algunas definiciones que serán empleadas en la explicación posterior.

Definición 34 *Los nodos de carga-descarga (NCD) son aquellos nodos que representan una posición Ψ en C_f donde el vehículo puede ser cargado o descargado.*

Definición 35 *Se le llamará salones a todas aquellas áreas limitadas por paredes y entradas, tales como cuartos, oficinas, salas, etc.*

Definición 36 *Los nodos de enlace son los nodos que representan las entradas de un salón.*

Definición 37 *Los arcos de enlace son los arcos que unen a dos nodos de enlace.*

De acuerdo con las definiciones anteriores los nodos que representan a las configuraciones q_{ini} y q_{fin} son en principio NCD, ya que en el problema básico del AGV, el transporte de piezas de una configuración a otra implica que tanto q_{ini} como q_{fin} sean nodos NCD. Sin embargo en C_f puede haber varios puntos de carga y descarga de piezas, lo que resulta conveniente para dar mayor flexibilidad al sistema. En el simulador se pueden definir estos puntos situando NCD's en las posiciones correspondientes.

En un ambiente de oficinas, como en el que se pretende trabajar el AGV simulado, se tienen diferentes salones o cuartos con puertas, que pueden formar complicadas regiones C-obstáculo si consideramos a las paredes como parte de ella, ya que esto daría lugar a obstáculos poligonales muy extensos. Para simplificar esto, el espacio C_f se divide en salones en los cuales se sitúan dos nodos de enlace en las puertas, como lo muestra la figura 3.5, para representar el paso de un salón a otro. La distancia D_n es la distancia de separación entre los nodos y es un parámetro que se puede ajustar dependiendo de la longitud del vehículo particular, de manera que al cruzar la entrada, éste quede completamente dentro del salón. Así pues, se considera a cada salón como una región independiente que se une con un salón adyacente a través de los nodos de enlace unidos por sus arcos de enlace.

En el grafo de conectividad se pretende representar el espacio libre del AGV. Pensando una vez más en el problema básico del AGV de transportar piezas de un punto a otro, q_{ini} y q_{fin} respectivamente, dentro de C_f queda claro que: el vehículo tendrá que cruzar ciertas configuraciones en C_f que le permitan ir de q_{ini} a q_{fin} , y el grafo de conectividad permite

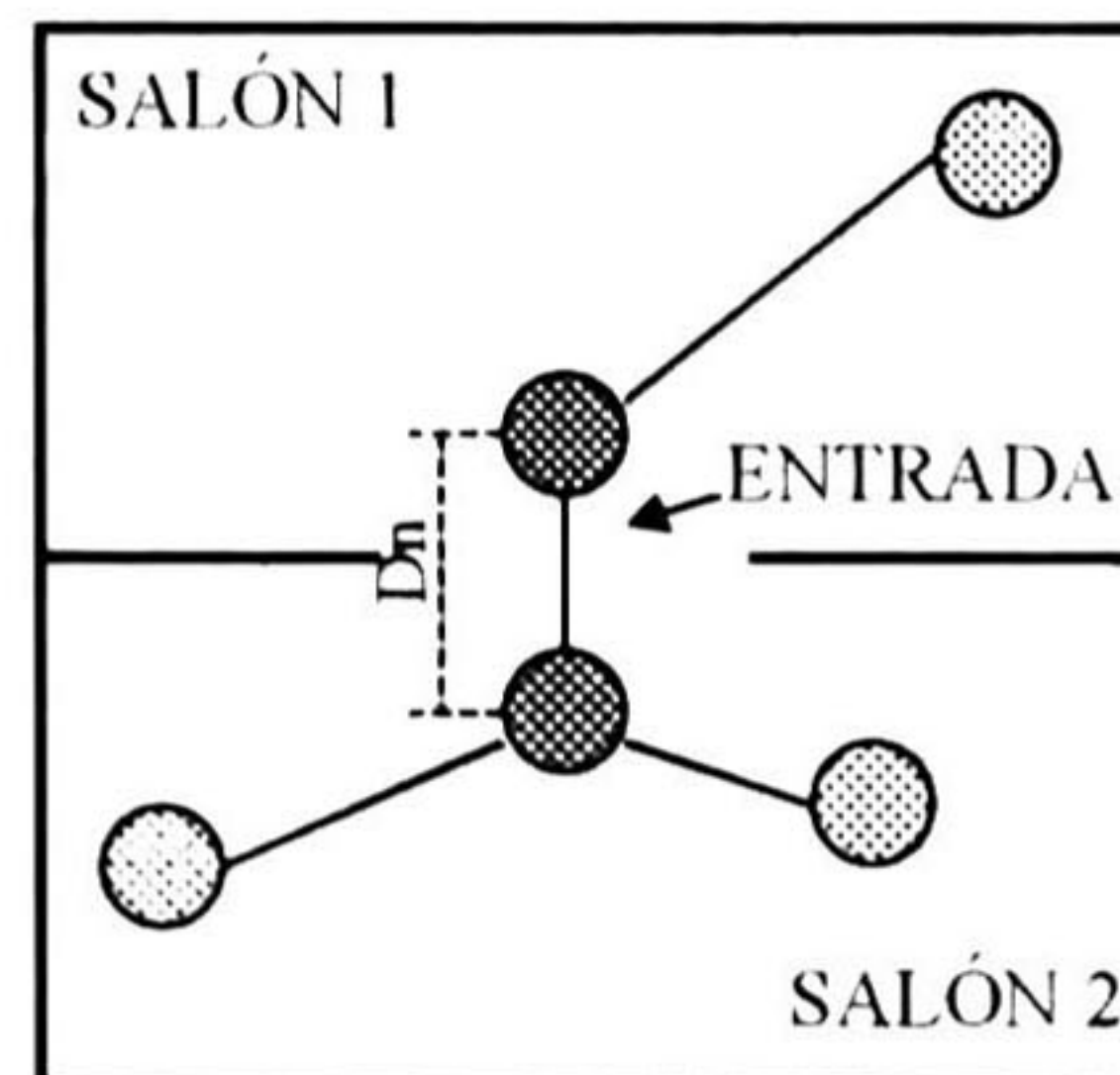


Figura 3.5: Nodos y arco de enlace de la entrada que divide a los salones 1 y 2.

encontrar la ruta que proporcione la información necesaria para formar la trayectoria. Pues bien, con el fin de simplificar el grafo de conectividad podemos establecer que se haga la descomposición de celdas solamente en aquellos salones en los que se necesite cruzar su interior. Esto queda establecido si proponemos la siguiente regla: *La descomposición de celdas se realizará si y solo si en el salón se tiene un NCD o si el salón tiene dos o más entradas.*

La explicación de la conveniencia de esta regla es sencilla. Primero, si tenemos un salón S_1 que no tiene NCD esto significa que no se ha definido una posición Ψ para entrega o recepción de piezas dentro de ese salón y en consecuencia el vehículo no tendrá que cruzarlo para cumplir con el problema básico. Así pues en caso de que se pida al vehículo que lleve una pieza desde algún punto de carga al salón S_1 , el vehículo solamente tendrá que cruzar la entrada de S_1 y esperar su descarga en el nodo de enlace más cercano dentro de S_1 . Por otro lado, si el salón S_2 tiene un NCD en su interior necesariamente el vehículo requiere mayor información del espacio libre dentro de S_2 ya que tendrá que cruzarlo para entregar o cargar piezas desde ese punto, por lo que la descomposición de celdas dentro de S_2 es necesaria. De igual forma, en aquellos salones que tienen dos o más entradas se necesita hacer una descomposición de celdas, esto se debe a que el vehículo puede determinar que es conveniente cruzar este salón, entrando por una de sus entradas y saliendo por otra para obtener una ruta más corta. Así pues, en los casos en que el vehículo tenga que cruzar por el interior de un salón necesita contar con más información de C_f correspondiente al salón y entonces la descomposición de celdas se vuelve necesaria. De otra manera, la descomposición de celdas no es necesaria y se puede evitar. Por ejemplo, en los salones 2 y 4 de la figura 3.6 no se realiza la descomposición de celdas, en el salón 1 se encuentra un NCD, marcado con un círculo, por lo que se hace la descomposición en su interior. En los salones 3 y 5 se también hay NCDs y además tienen más de dos entradas. Las líneas punteadas en las entradas de los salones indican el “bloqueo” de las entradas para evitar que las celdas de los salones se extiendan hacia afuera de ellos.

Si un salón S requiere la descomposición de celdas en su interior, ésta se realizará como lo indica el algoritmo de la sección anterior, sin embargo se debe hacer la observación siguiente:

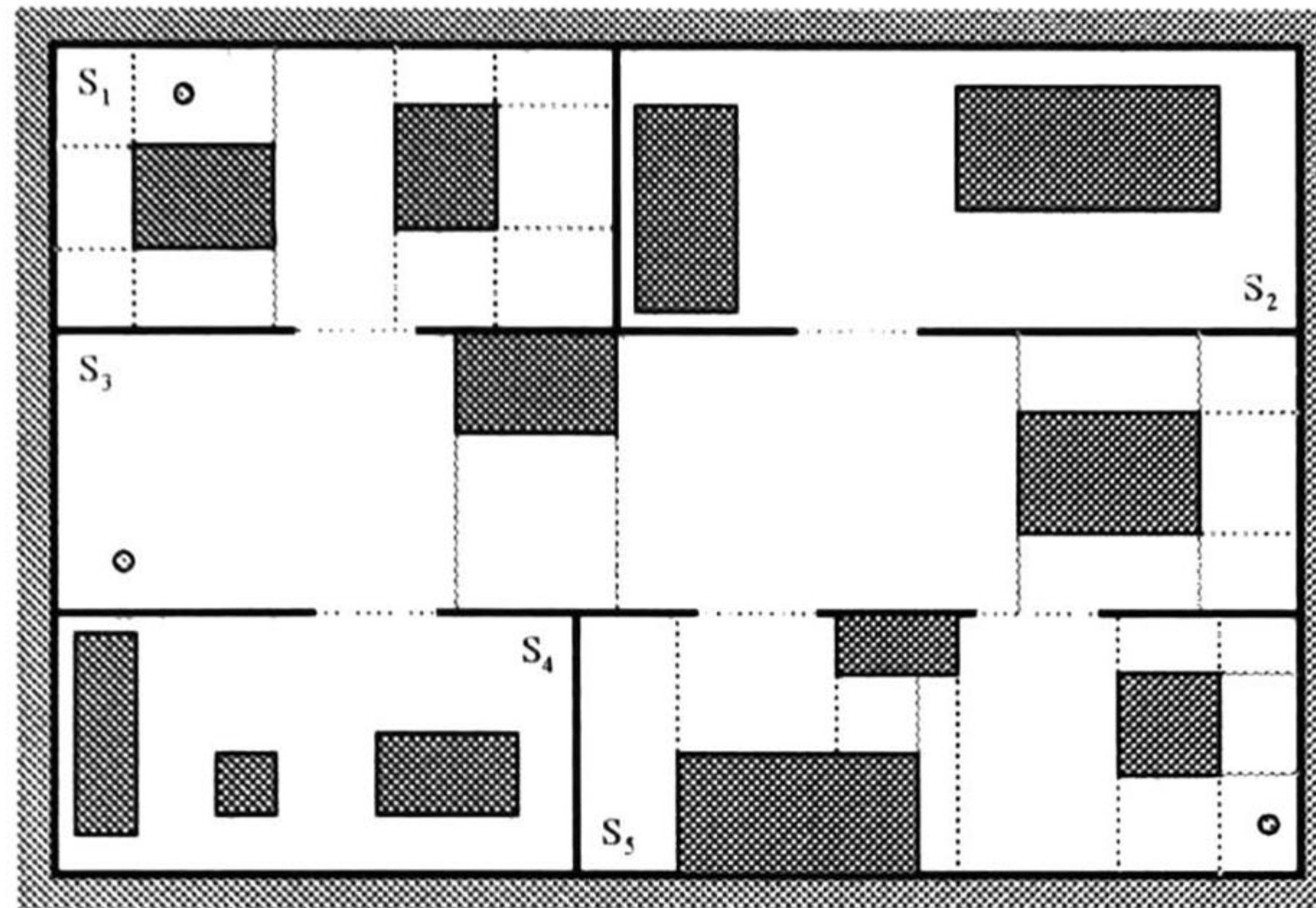


Figura 3.6: La figura muestra una descomposición de celdas de un espacio de trabajo del vehículo.

Las entradas del salón se “bloquearán” para formar los límites del salón. Para bloquearlas considere que se tienen puertas y que estas están cerradas. (Ver la figura 3.6).

Una vez hechas las aclaraciones anteriores se pueden establecer las especificaciones para generar el grafo de conectividad. El procedimiento para generarlo es el siguiente:

1. Identificar los salones del espacio de trabajo.
2. Asignar los nodos de enlace con sus arcos de enlace en las entradas de los salones.
3. Localizar los puntos de carga-descarga en el espacio de trabajo y asignar un NCD en la posición correspondiente.
4. Realizar la descomposición de celdas trapezoidal en los salones que tengan en su interior al menos un NCD o que tengan al menos dos entradas.
5. Generar el grafo de conectividad de los salones que tuvieron una descomposición de celdas con las siguientes especificaciones:

- Los nodos del grafo de conectividad incluyen todos los nodos de enlace dentro del salón, todos los NCD dentro del salón y todos los nodos generados bajo los criterios siguientes:
 - Se asignará un nodo a los puntos medios de los segmentos L que limitan las celdas si y solo si la longitud del segmento dL cumple con: $Lmin < dL < Lmax$. Donde $Lmin$ y $Lmax$ son las longitudes mínima y máxima permitidas respectivamente. Estas distancias se especifican de acuerdo a parámetros del vehículo particular.
 - Si $dL < Lmin$ no se asignará nodo al segmento L . Si L es un segmento horizontal originado por un vértice exterior X_t , los segmentos verticales originados por X_t se

48CAPÍTULO 3. REPRESENTACIÓN DEL MAPA Y BÚSQUEDA DE LA RUTA ÓPTIMA.

eliminarán formándose celdas externas de mayor área. Estas celdas formadas por dos celdas anteriores pueden ser celdas externas de doble segmento, en cuyo caso se deben dividir por medio de segmentos horizontales como se indica en la sección anterior (Ver figura 3.7). Si L es un segmento vertical originado por un vértice exterior X_t , el segmento horizontal originado por X_t se eliminará formándose una celda externa de mayor área (Ver figura 3.7).

- Si $dL > L_{max}$ se asignará un número $N_{\mathcal{L}\mathcal{L}}$ de nodos en el segmento L , dado por $NNL = \text{int}\left(\frac{dL}{L_{div}}\right)$, donde L_{div} es la *distancia de división* especificada de acuerdo a parámetros del vehículo particular, **int** representa la función **integer** que devuelve la parte entera del argumento. Los NNL nodos se distribuyen a lo largo del segmento L . Para ubicar los NNL nodos en L , se divide el segmento L en los trozos de segmento Lp_i $i=1, \dots, NNL$, de igual longitud dada por $dLp = \frac{NNL}{dL}$. Los nodos se ubicarán en los puntos medios de cada segmento Lp_i $i=1, \dots, NNL$. (Ver la figura 3.8).

- Se deben etiquetar los nodos para identificarlos en el grafo. Los nodos se pueden enumerar consecutivamente mientras son formados de acuerdo a los criterios anteriores.
- Los arcos del grafo de conectividad son arcos no dirigidos que se forman por líneas rectas que unen dos nodos que pertenecen a la misma celda, con las excepciones siguientes:
 - Cuando los nodos pertenezcan a dos segmentos L diferentes los cuales se hayan originado por un mismo vértice X o por dos vértices diferentes alineados verticalmente (Ver la figura 3.9).
 - Cuando los nodos pertenezcan a una celda cuya longitud horizontal d_x sea menor a d_{min} , donde d_{min} es un parámetro especificado para un vehículo específico, se deben considerar los casos siguientes:
 - * Si la celda es una celda exterior, está formada por segmentos L sólo de un lado de la celda; los nodos correspondientes al segmento L se suprimirán del grafo.
 - * Si la celda no es exterior, está formada por segmentos L a la izquierda y derecha, que llamaremos *segmentos críticos izquierdo y derecho*, respectivamente. Los segmentos críticos son originados por *vértices críticos* de los C-obstáculos. Solamente dos vértices de un mismo C-obstáculo pueden ser críticos. Si dos vértices críticos pertenecen a un mismo C-obstáculo y además están alineados verticalmente formando segmentos L alineados, los llamaremos *par crítico*, representado por X_1-X_2 , donde el vértice X_1 está arriba del vértice X_2 . Sea la distancia $PL = \sqrt{d_x^2 + d_y^2}$ la *distancia crítica*, donde la distancia d_y es la distancia paralela al eje “y” entre los vértices críticos X_{c1} y X_{c2} . Los arcos se eliminarán si se cumple alguno de los casos siguientes:

Si dos vértices críticos, X_1 y X_2 , no están alineados verticalmente y no forman un par crítico con algún otro y además pertenecen a diferente C-obstáculo, entonces cada vértice forma un segmento crítico L hacia arriba y otro hacia abajo. Se calcula la distancia PL entre los vértices X_1 y X_2 . Si $PL < d_{min}$, se eliminan los arcos entre los nodos correspondientes al segmento crítico arriba de X_1 con los nodos correspondientes al segmento crítico abajo de X_2 y viceversa. (Ver la figura 3.10).

Si la ordenada de un vértice crítico X_3 está entre las ordenadas de un par crítico, X_4-X_5 , se eliminan los arcos entre los nodos correspondientes al segmento crítico arriba de X_3 con los nodos correspondientes al segmento crítico abajo de X_5 y los arcos entre los nodos correspondientes al segmento crítico abajo de X_3 con los nodos correspondientes al segmento crítico arriba de X_4 . (Ver la figura 3.10).

Si la ordenada de algún vértice del par crítico X_6-X_7 está entre las ordenadas de otro par crítico, X_8-X_9 , se eliminan los arcos entre los nodos correspondientes al segmento crítico arriba de X_6 con los nodos correspondientes al segmento crítico abajo de X_9 y los arcos entre los nodos correspondientes al segmento crítico abajo de X_7 con los nodos correspondientes al segmento crítico arriba de X_8 . (Ver la figura 3.10).

Si la ordenada del vértice X_{10} del par crítico $X_{10}-X_{11}$, que forma un segmento L , es menor a la ordenada del vértice X_{13} del par crítico $X_{12}-X_{13}$, que forma el segmento opuesto se calcula la distancia PL entre los vértices X_{10} y X_{13} . Si $PL < d_{min}$ se eliminan los arcos entre los nodos correspondientes al segmento crítico arriba de X_{10} con los nodos correspondientes al segmento crítico abajo de X_{13} . (Ver la figura 3.10).

Si la ordenada del vértice X_{14} de un par crítico $X_{14}-X_{15}$ que forma un segmento L es menor a la ordenada del vértice crítico X_{16} que no forma par crítico, se calcula la distancia PL entre X_{14} y X_{16} . O bien, si la ordenada del vértice X_{15} es mayor a la ordenada del vértice crítico X_{17} que no forma par crítico, se calcula la distancia PL entre X_{15} y X_{17} . Si $PL < d_{min}$ se eliminan los arcos entre los nodos correspondientes a los segmentos críticos arriba de X_{14} y abajo de X_{16} , para el primer caso, o los arcos entre los nodos correspondientes a los segmentos críticos abajo de X_{15} y arriba de X_{17} , para el otro caso. (Ver la figura 3.10).

- Se asigna el peso correspondiente a los arcos del grafo. El peso de un arco corresponde a la distancia lineal entre los nodos que une.

La figura 3.7 representa los casos cuando la distancia $dL < L_{min}$. En la figura (a) la distancia d_1 corresponde a segmentos horizontales L cuya longitud $dL < L_{min}$ por lo que son eliminados los segmentos verticales de los vértices correspondientes formando una celda

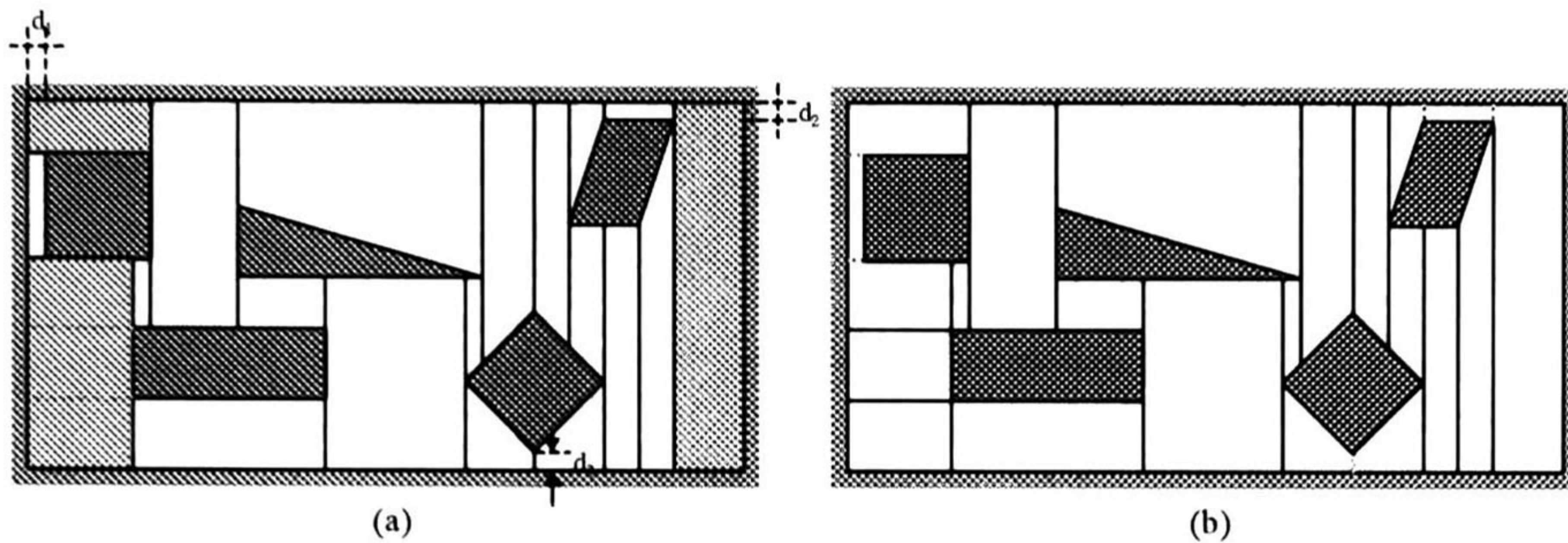


Figura 3.7: Distancia $dL < Lmin$.

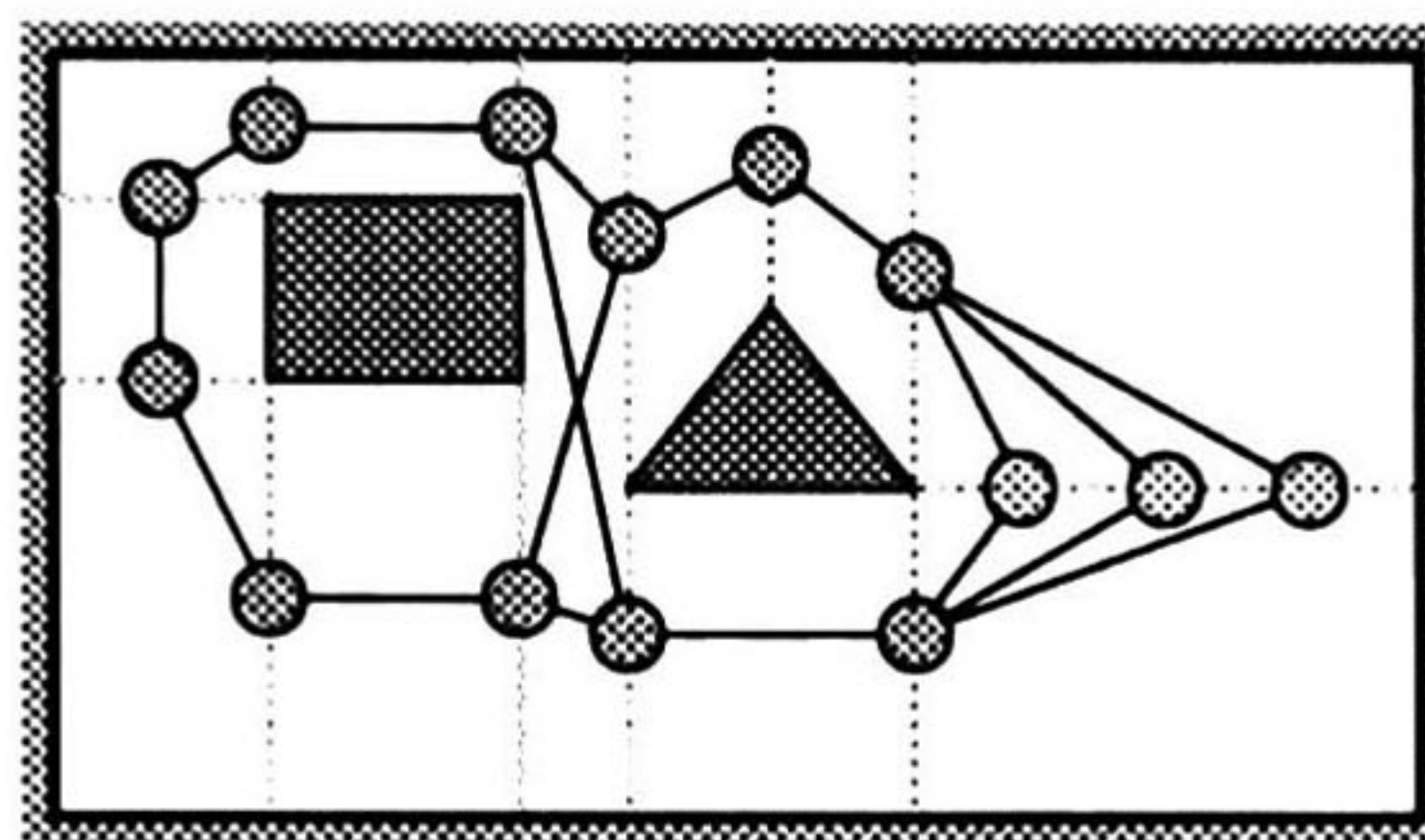


Figura 3.8: Distancia $dL > Lmax$.

mayor. La celda inferior izquierda generada es una celda exterior de doble segmento, por lo que debe dividirse trazando los segmentos horizontales punteados. La distancia d_2 corresponde a un segmento vertical de una celda exterior de doble segmento, por lo que el segmento horizontal del vértice correspondiente se elimina formando una celda mayor. La distancia d_3 , que corresponde a un segmento vertical, solamente provoca que no se asigne nodo al segmento correspondiente. En la figura (b) se muestra la descomposición de celdas resultante.

En la figura 3.8, se muestra un caso donde el segmento L horizontal tiene una distancia $dL > Lmax$, por lo que se asignan en él tres nodos.

Las figuras 3.9 y muestran los casos en que se suprimen los arcos entre nodos que corresponden a una misma celda. En la figura 3.9, los nodos más claros en la figura corresponden a nodos de una celda común, formando dos grupos para las dos celdas correspondientes. Los nodos superiores de estos grupos solamente se unen mediante un arco con los nodos inferiores que se encuentran en el lado opuesto de la celda, sin asignarse arco que una a un nodo superior con un nodo inferior que estén alineados verticalmente. En la figura 3.10 se muestran los casos más comunes en que la distancia horizontal d_x de una celda es menor a $dmin$, ocasionando que se supriman los arcos entre ciertos nodos de los segmentos L formados por los vértices críticos involucrados.

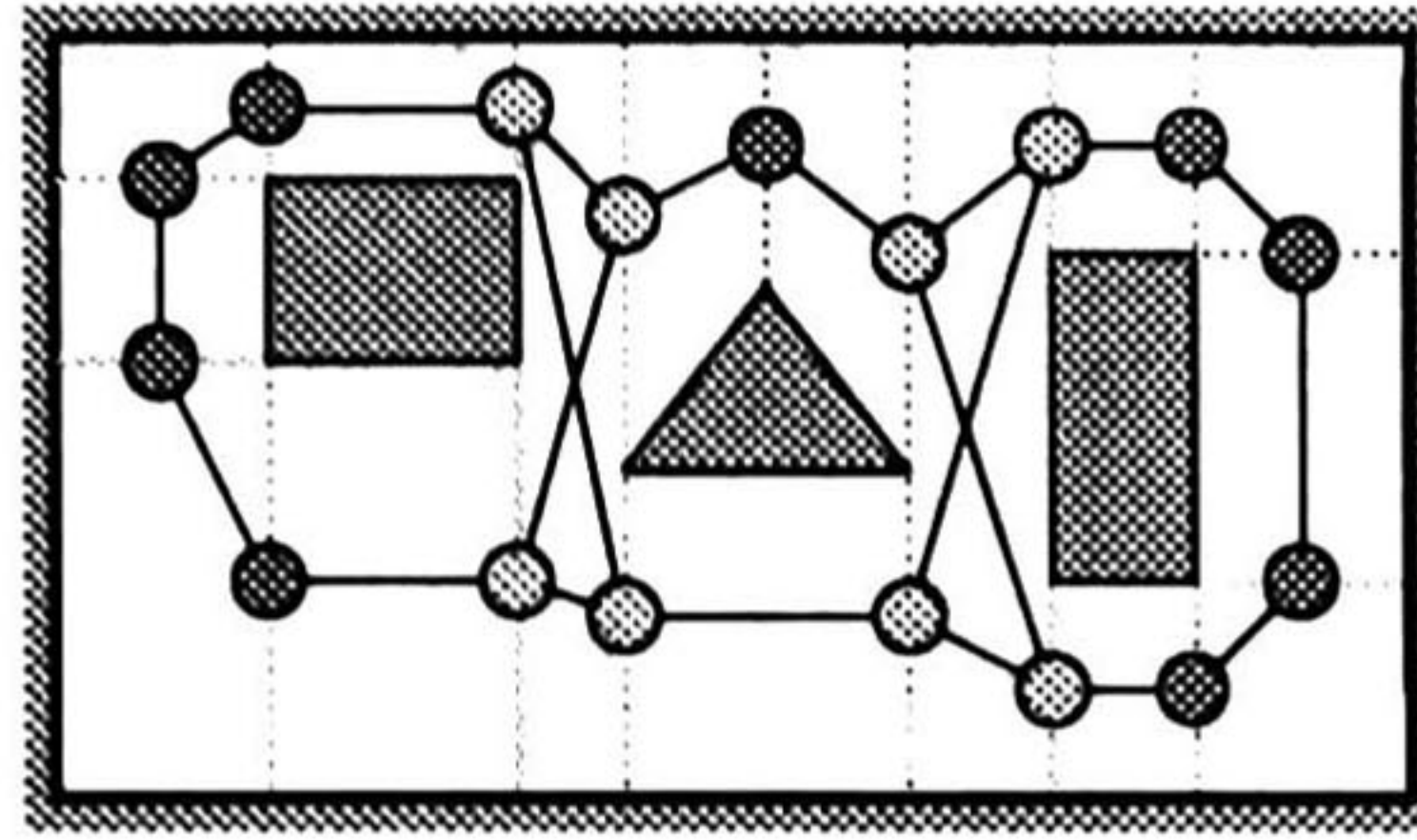


Figura 3.9: Omisión de arcos entre nodos de una misma celda.

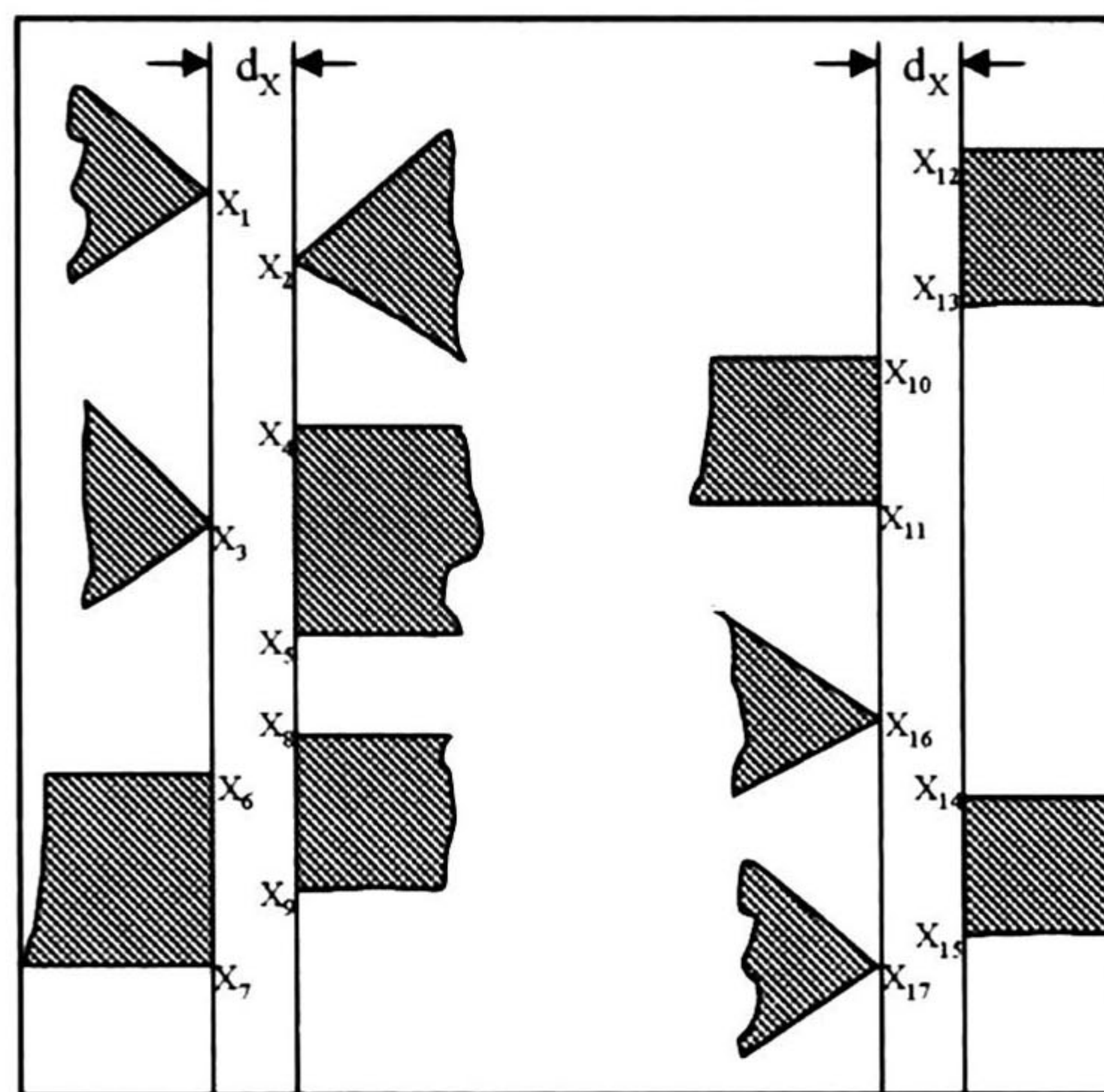


Figura 3.10: Distancia $dx < dmin$.

Antes de mostrar un ejemplo, debemos especificar cómo determinar los parámetros siguientes: $Lmin$, $Lmax$, $Ldiv$, $dmin$ y Dn . $Lmin$ es la distancia mínima permitida de un segmento L para asignarle un nodo, y debido que al asignar un nodo en el segmento L se asume que el vehículo cruzará por el nodo atravesando por la sección del segmento L , por lo tanto, $Lmin$ debe ser al menos la longitud del ancho del vehículo. $Lmax$ es la distancia máxima del segmento L antes de asignarle más de un nodo. La razón de asignar dos o más nodos a un segmento L es generar rutas más cortas para el vehículo al cruzar el límite L de una celda. Así que $Lmax$ debe ser el límite donde ya no sea costeable tener un solo nodo en la mitad de L en una ruta que cruce ese segmento. Y $Ldiv$ es la distancia especificada entre dos nodos de un mismo segmento L . La distancia $dmin$ corresponde la distancia horizontal mínima permitida para una celda. Considerando que el vehículo pueda requerir cruzar la celda en forma vertical, $dmin$ puede calcularse con el mismo criterio que $Lmin$. Dn es la distancia entre los nodos de enlace y este parámetro debe especificarse de manera que el vehículo tenga un buen desempeño al cruzar las entradas de los salones y, como estos nodos de enlace pueden considerarse como nodos NCD en salones sin descomposición de celdas, se debe tomar en cuenta el largo del vehículo para que quede totalmente dentro del salón en el

52CAPÍTULO 3. REPRESENTACIÓN DEL MAPA Y BÚSQUEDA DE LA RUTA ÓPTIMA.

momento de ser cargado o descargado.

Una vez establecidos los criterios para generar el grafo de conectividad, mostraremos un ejemplo de generación del grafo correspondiente a un espacio de trabajo del vehículo. En el ejemplo siguiente se muestra la información inicial requerida y la formación paso a paso del grafo de conectividad correspondiente. Para el ejemplo considere el mapa de la figura 3.11.a que corresponde a un espacio de trabajo para el AGV y para el cual queremos generar su grafo de conectividad.

Primero se deben identificar los salones para localizar sus entradas. En las entradas de los salones se asignan los nodos de enlace con sus respectivos arcos de enlace. Se identifican también los nodos NCD que se especifican en C_f . Estos tres pasos se ven simultáneamente en la figura 3.11.b. Es importante identificar aquellos salones que requieren una descomposición de celdas de su espacio libre, en el ejemplo les corresponde a los salones 1, 2, 4 y 6. La descomposición se muestra en la figura 3.11.c. Ahora se asignan los nodos a los segmentos L de las celdas formadas, recordando las reglas de omisión. Los nodos asignados a los segmentos L de las celdas se muestran en la figura 3.11.d. Los nodos formados deben etiquetarse para ser identificados, sin embargo, este paso se omite en el ejemplo para evitar que se complique la figura. Las etiquetas que se deben asignar a los nodos pueden ser números, letras o nombres, lo importante es que sean únicas para cada nodo para su perfecta identificación. Una vez que contamos con todos los nodos que componen el grafo de conectividad, se forman los arcos. Para asignar un arco entre dos nodos se deben tener en cuenta los casos de omisión mencionados en las especificaciones. Los arcos formados se muestran en la figura 3.11.e. El último paso es asignar a cada arco su peso correspondiente a la distancia entre los nodos que une. Este paso también es omitido en el ejemplo mostrado. La distancia que se asigna al peso de los arcos es la distancia entre las posiciones reales en C_f de la posición de los nodos, que quedan representados en el mapa a escala de C_f . Finalmente, en la figura 3.11.f se muestra el grafo de conectividad generado que representa el espacio libre en el mapa.

El algoritmo descrito en el ejemplo anterior es un poco diferente al planteado en un principio, sin embargo este algoritmo no deja de ser sencillo y simple. Pueden inclusive, buscarse otras situaciones que puedan ser igualmente importantes en la planeación de la ruta con el fin de hacer otras modificaciones en el algoritmo, sin embargo se debe cuidar la simplicidad del algoritmo pensando siempre en la complejidad de la implementación. El algoritmo de generación del grafo de conectividad se resume en el Apéndice 1. Este algoritmo no está implementado en el programa del simulador, sin embargo éste se puede aplicar teniendo la información del espacio de trabajo del vehículo y se puede introducir el grafo en el simulador mediante el procedimiento descrito en el Apéndice 2.

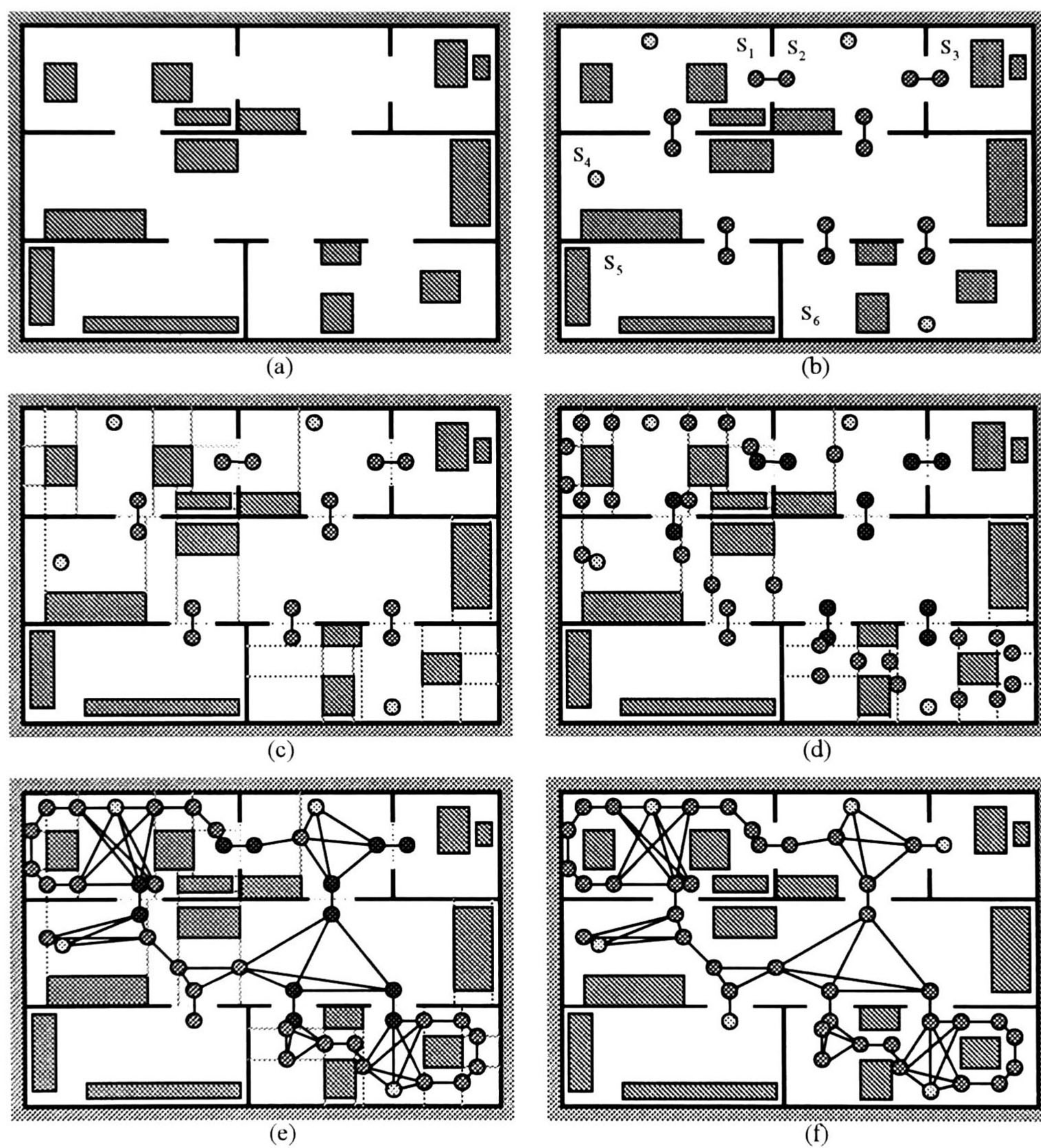


Figura 3.11: Ejemplo de generación de un grafo de conectividad.

3.2 BÚSQUEDA DE LA RUTA ÓPTIMA.

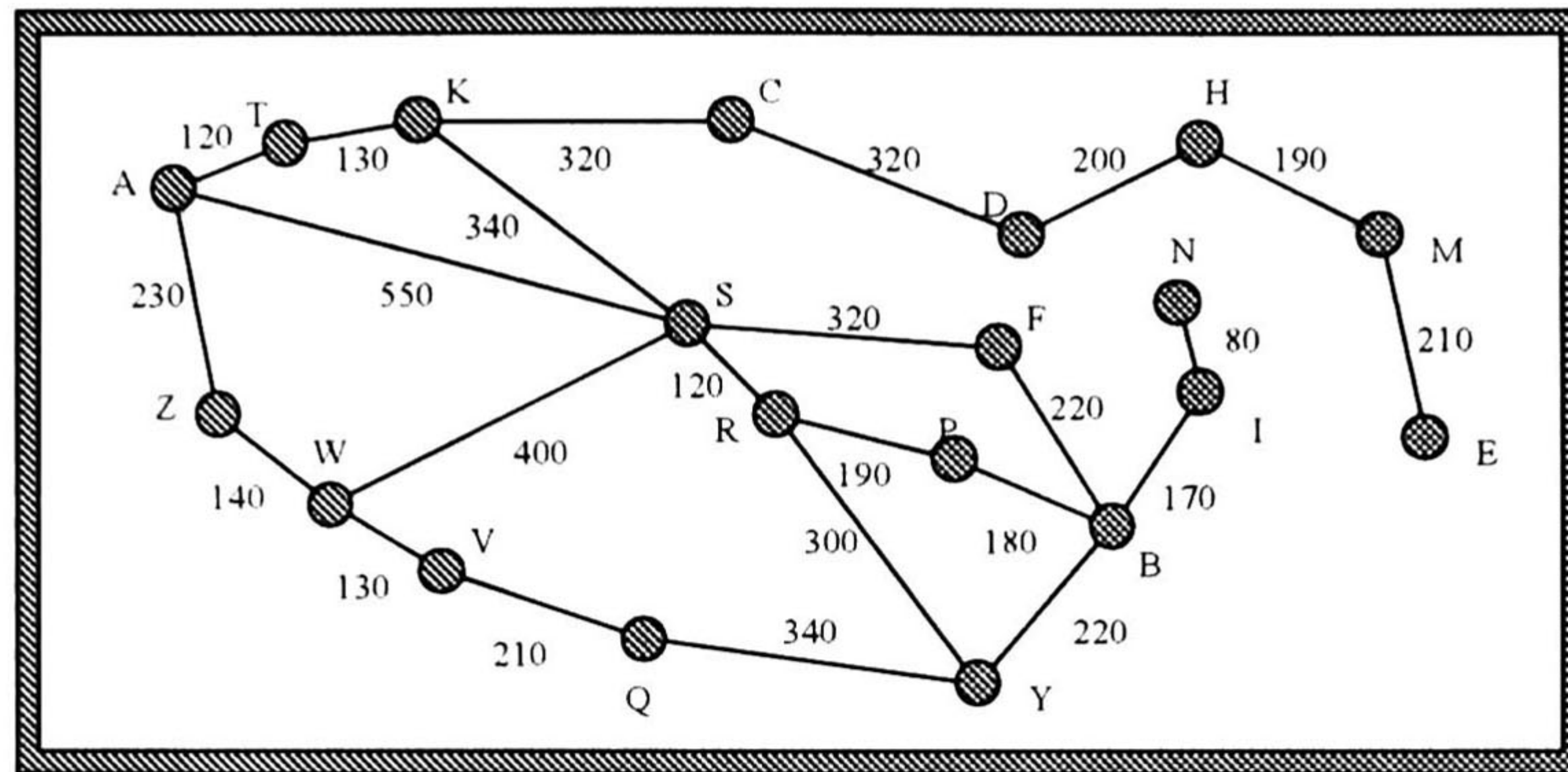
Como se mencionó anteriormente, la ruta que se busca en el grafo de conectividad, determinado por el método de descomposición de celdas de la sección anterior, es la ruta más corta en el grafo de conectividad que conecta las configuraciones inicial y final. Así pues, *la ruta buscada es óptima en distancia.*

La búsqueda de la ruta óptima en el simulador del AGV de esta tesis emplea un algoritmo de IA *informado*. En el capítulo anterior se mencionó el empleo de una función de evaluación que permite determinar el mejor prospecto a ser expandido en la búsqueda de la ruta óptima. Además se mencionó la implementación de una función heurística que ayuda también a determinar cuál es el mejor prospecto a expandirse.

La implementación de una función de evaluación que nos permita emplear el conocimiento que tenemos a priori del espacio de trabajo del vehículo, para buscar la ruta más corta, forma parte de una estrategia denominada *búsqueda del mejor primero*. En la búsqueda del mejor primero se ordenan los nodos de manera que aquel con mejor evaluación se expande primero. Existe toda una familia de algoritmos de búsqueda del mejor primero que emplean diferentes funciones de evaluación. Debido a que estos métodos proponen encontrar soluciones con el menor costo, estos algoritmos típicamente usan alguna medida estimada del costo de la solución y tratan de minimizarla. Una de estas medidas es el costo de la trayectoria en el grafo denotada como $g(n)$, donde esta función mide el costo de la trayectoria desde el nodo inicial al nodo n . El problema que se presenta considerando esta función de evaluación es que algunas veces la función puede desviar la búsqueda. Para poder dirigir la búsqueda enfocándola hacia la solución, la medida de la función de evaluación debe incorporar algún estimado del costo de la trayectoria desde un nodo al nodo más cercano a la solución.

Existe una estrategia de búsqueda denominada *voráz* (“*Greedy search*”), la cual emplea la misma estrategia del mejor primero minimizando el costo estimado para alcanzar la meta. Es decir, que se expande primero aquel nodo cuyo estado se determina que es más cercano al estado de la solución. Sin embargo, con este método “*para la mayoría de los problemas, el costo para alcanzar la solución desde un estado particular se puede estimar pero no se puede determinar exactamente*” [19]. La función que calcula el costo estimado para alcanzar el estado solución es una *función heurística*, denotada por $h(n)$ y que representa el costo estimado de la trayectoria más corta desde el estado del nodo n al estado solución. La función h puede ser en general cualquier función, el único requerimiento es que $h(n)=0$ si n es la solución.

La función heurística apropiada depende del problema específico. Para el problema que tenemos de encontrar la ruta de una configuración inicial a una configuración final, se puede determinar una función heurística que sea la distancia en línea recta del nodo n a la configuración final. Esto es:



Distancias en línea recta de cada nodo al nodo B.			
A	1000	F	220
B	0	H	380
C	550	I	170
D	300	K	800
E	340	M	400
		N	230
		P	180
		Q	490
		R	360
		S	470
		T	920
		V	670
		W	780
		Y	220
		Z	910

Figura 3.12: Grafo ponderado con las distancias entre nodos.

$$h(n) = \text{Distancia en línea recta entre } n \text{ y la configuración final.}$$

En el grafo mostrado en la figura 3.12, los pesos de los arcos corresponden a las distancias entre los nodos que unen y la tabla especifica las distancias en línea recta de cada nodo al nodo B. En éste grafo podemos mostrar el progreso de una búsqueda voráz para una ruta entre los nodos A y B. Iniciando en el nodo A, el primer nodo que se expande desde A es S, ya que es el más cercano a B en comparación con Z o T. El siguiente nodo expandido será F, porque está más cercano a B. El nodo F a su vez generará a B, que es la meta. Para este ejemplo particular, la función heurística falló en la determinación de una trayectoria de costo mínimo, ya que nunca expandió otro nodo que no estuviera en la trayectoria solución encontrada. Como se puede apreciar, esta solución no es óptima porque la ruta A, S, F, B tiene una mayor distancia que la ruta A, S, R, P, B. Sin embargo esta ruta óptima no fue encontrada porque F está más cerca de B en una distancia en línea recta que R de B, así que F fue expandido primero. Se puede decir que esta estrategia prefiere “devorar” lo antes posible la distancia restante a la meta, sin considerar si esta será la mejor decisión a la larga, de aquí viene el adjetivo de “voráz”. Esta estrategia tiende a buscar la solución rápidamente, aunque en algunos casos, como el anterior, no siempre se encuentra la solución óptima.

Consideremos ahora que queremos encontrar una ruta de I a F en la misma figura. El algoritmo expandirá primero N, que no tiene salida. En este caso el algoritmo expande nodos innecesarios y aún más, si no se tiene cuidado en este tipo de situaciones el algoritmo puede ciclarse y nunca encontrar la solución, la búsqueda se quedaría oscilando entre I y N.

56CAPÍTULO 3. REPRESENTACIÓN DEL MAPA Y BÚSQUEDA DE LA RUTA ÓPTIMA.

Los métodos de búsqueda mejor primero y búsqueda voráz presentan el problema de no ser óptimos ni completos. Sin embargo existen los llamados *algoritmos de búsqueda admisible* y son aquellos que garantizan encontrar una ruta de costo mínimo de un nodo cualquiera al nodo meta siempre que exista tal ruta. Estas estrategias son en principio búsquedas mejor primero empleando una función de evaluación $f(n)$ definida como la suma de las funciones $g(n)$ y $h(n)$. Para formalizar estas estrategias daremos las siguientes definiciones [19, 14]:

Definición 38 *Considerando la función de evaluación $f(n)=g(n)+h(n)$, donde n es cualquier estado encontrado en la búsqueda, $g(n)$ es el costo de n desde el estado inicial, $h(n)$ es una estimación heurística del costo de ir de n al estado solución. Si esta función de evaluación es empleada para determinar el siguiente nodo expandible en el algoritmo de búsqueda mejor primero, el resultado es llamado algoritmo A.*

Definición 39 *Un algoritmo de búsqueda es admisible si, para cualquier grafo, éste siempre termina con la ruta de solución óptima siempre que exista una ruta del estado inicial al estado solución.*

Definición 40 *Si la función heurística $h(n)$ es tal que nunca sobre-estima el costo para alcanzar la meta, se dice que es una función heurística admisible.*

Definición 41 *Si el algoritmo A es usado con una función de evaluación $f(n)=g(n)+h(n)$, tal que $h(n)$ es una función heurística admisible, el algoritmo de búsqueda resultante se denomina algoritmo A^**

La sección siguiente muestra las características del algoritmo A^* , que es el empleado en la búsqueda de la ruta óptima en el grafo de conectividad del simulador desarrollado.

3.2.1 ALGORITMO A^* .

El algoritmo A^* es pues, un algoritmo de búsqueda mejor primero que emplea como función de evaluación a $f(n)=g(n)+h(n)$. El algoritmo A^* es óptimo y es completo. Antes de probar estas características que lo hacen adecuado para nuestro problema de encontrar una ruta óptima en el grafo de conectividad, es conveniente conocer la manera en que trabaja este algoritmo.

Como se mencionó anteriormente el algoritmo A^* requiere una función heurística admisible para garantizar una solución óptima en la búsqueda si ésta existe, la función $h(n)$ propuesta previamente como la distancia en línea recta de un nodo al nodo meta es admisible. En el ejemplo de la figura 3.12 note que el algoritmo A^* prefiere expandir el nodo R en lugar del nodo F cuando buscamos una ruta de A hacia B. Esto es porque aún cuando F es más cercano a B, la ruta por F no es tan eficiente en llegar a B por el camino más corto

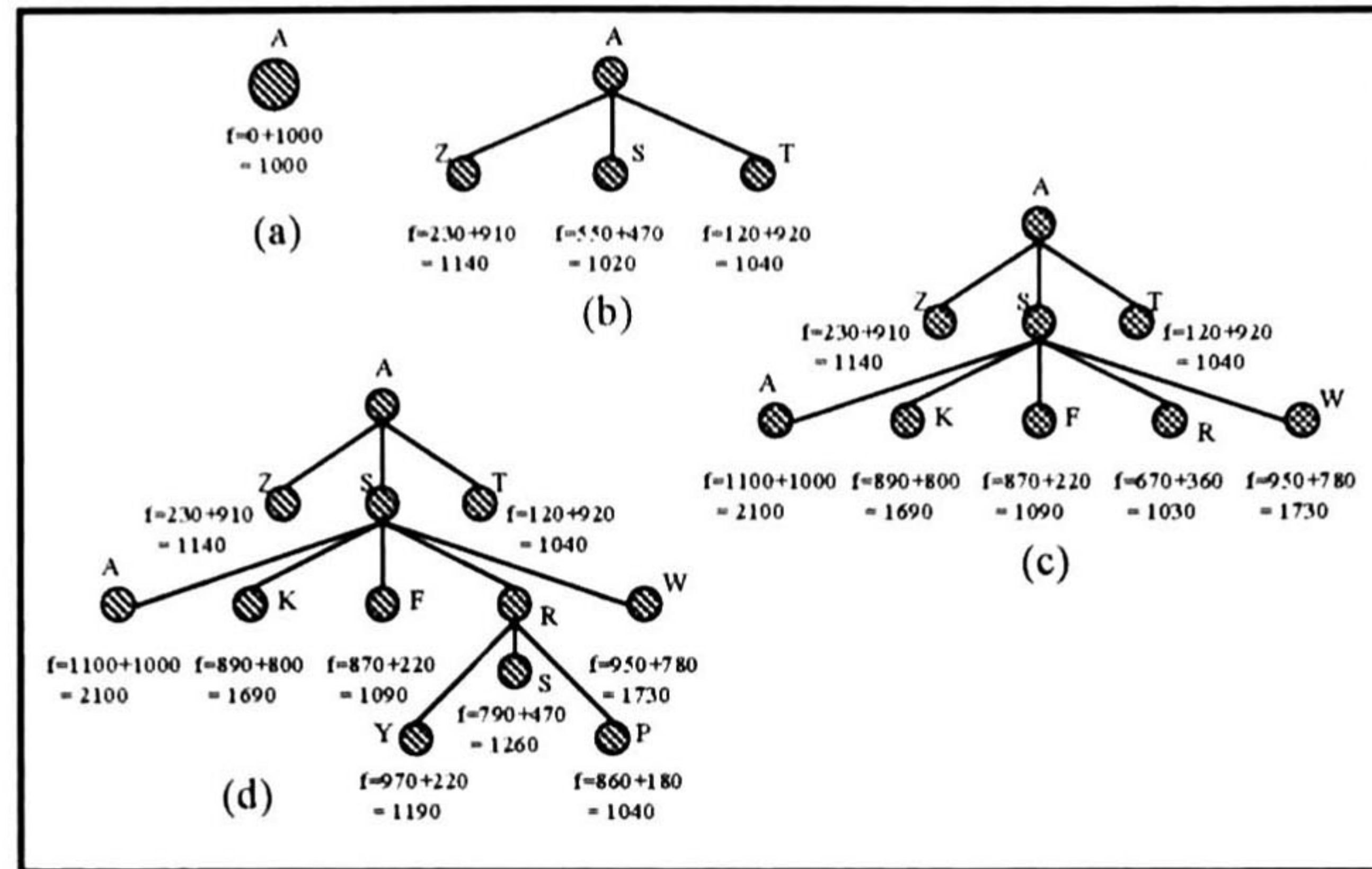


Figura 3.13: Expansión sucesiva de nodos.

como lo es la ruta por R. La figura 3.13 muestra la expansión de los nodos en la búsqueda del A*.

En la figura 3.13 se puede observar la característica de *monotonidad* de la función $f(n)$, es decir, la función $f(n)$ nunca decrece durante el recorrido de cualquier trayectoria desde el nodo raíz. Esta función se dice que es *monótona* y se mantiene esta propiedad para casi cualquier función heurística admisible. Para cualquier función heurística no-monótona se puede hacer un ajuste que permita recuperar la propiedad de monotonidad de la función. Si consideramos dos nodos n_1 y n_2 , donde n_1 es el único padre de n_2 y suponiendo que tenemos $g(n_1)=3$ y $h(n_1)=4$. Entonces $f(n_1)=7$, y con esto sabemos que el costo de una ruta solución a través de n_1 es al menos de 7. Ahora supongamos que n_2 es sucesor de n_1 y que $g(n_2)=4$ y $h(n_2)=2$, entonces $f(n_2)=6$. Este es un ejemplo de una función no-monótona. Sin embargo considerando el hecho de que cualquier ruta que pasa por n_2 pasa también por n_1 , podemos observar que el valor de 6 no tiene significado, ya que sabíamos de antemano que una ruta por n_1 tiene un costo al menos de 7. Entonces, verificando cada vez que se genera un nuevo nodo, si su costo f es menor que el costo f de su padre; si es así, se debe usar el costo f del padre, es decir:

$$f(n_2) = \max(f(n_1), g(n_2) + h(n_2)).$$

Esta función se llama *ecuación de trayectoria máxima (pathmax)*. De esta manera se eliminan los valores erróneos de las funciones heurísticas no-monótonas y f siempre será no-decreciente a lo largo de cualquier ruta desde la raíz, siendo h admisible.

El algoritmo A* implementado es el siguiente:

1. Se crean dos listas: ABIERTO y CERRADO.
2. Poner el nodo inicial en ABIERTO.

58CAPÍTULO 3. REPRESENTACIÓN DEL MAPA Y BÚSQUEDA DE LA RUTA ÓPTIMA.

3. Si ABIERTO está vacía, reportar falla.
4. Sacar el nodo V con el mínimo valor de la función de evaluación f de ABIERTO y ponerlo en CERRADO
5. Si V es la meta, reportar éxito.
6. Si V no es el nodo final o meta, se expande V enumerando cada uno de sus vértices adyacentes. Para cada vértice adyacente V' tomar una de las siguientes acciones:
 - a) Si V' no está en ABIERTO o CERRADO, ponerlo en ABIERTO; poner su apuntador a V ; y calcular sus valores de las funciones g y f .
 - b) Si V' está en ABIERTO, determinar si es que la trayectoria a través de V es más barata que la trayectoria actual a él. Si esto ocurre redireccionar su apuntador hacia V y recalcular los valores g y f . Esto corresponde a un ajuste del apuntador.
 - c) Si V' está en CERRADO y requiere ajuste de apuntador, sacarlo de CERRADO y regresarlo a ABIERTO.
7. Ir al paso número 3.

Al terminar la búsqueda, si se reportó *falla* entonces no existe una trayectoria del nodo inicial al nodo final, y si se reporta el final con *éxito* la ruta encontrada la forman los nodos de la lista CERRADO, la cual incluye todos los vértices expandidos en la búsqueda.

Es importante, como se ha visto, la función heurística en la búsqueda A^* . Existe una característica que nos da la noción de que una función heurística es de alguna manera mejor que otra para encontrar una ruta óptima cuando ambas son admisibles. Esta característica se presenta en la siguiente definición [14]:

Definición 42 Para dos funciones heurísticas admisible A^* , h_1 y h_2 , si $h_1(n) \leq h_2(n)$, para todos los estados n en el espacio de estados, la función heurística h_2 se dice que está más informada que h_1 .

La definición anterior señala que tanto h_1 y h_2 encontrarán la ruta óptima, pero h_2 evaluará menos estados en el proceso. En general, mientras más informado sea el algoritmo A^* , expandirá menos nodos en el espacio de estados para encontrar la solución óptima.

Al iniciar esta sección se mencionaron las características más importantes del algoritmo A^* , que son la optimidad y la completitud de este algoritmo. A continuación se probarán estas características.

PRUEBA DE QUE EL ALGORITMO ES ÓPTIMO.

Sea G un estado meta óptimo, y sea f^* el costo de la ruta solución óptima. Sea G_2 un estado meta subóptimo; esto es, un estado meta con un costo de ruta $g(G_2) > f^*$. Suponiendo que A^* selecciona a G_2 en la búsqueda. Debido a que G_2 es un estado solución, esto terminará

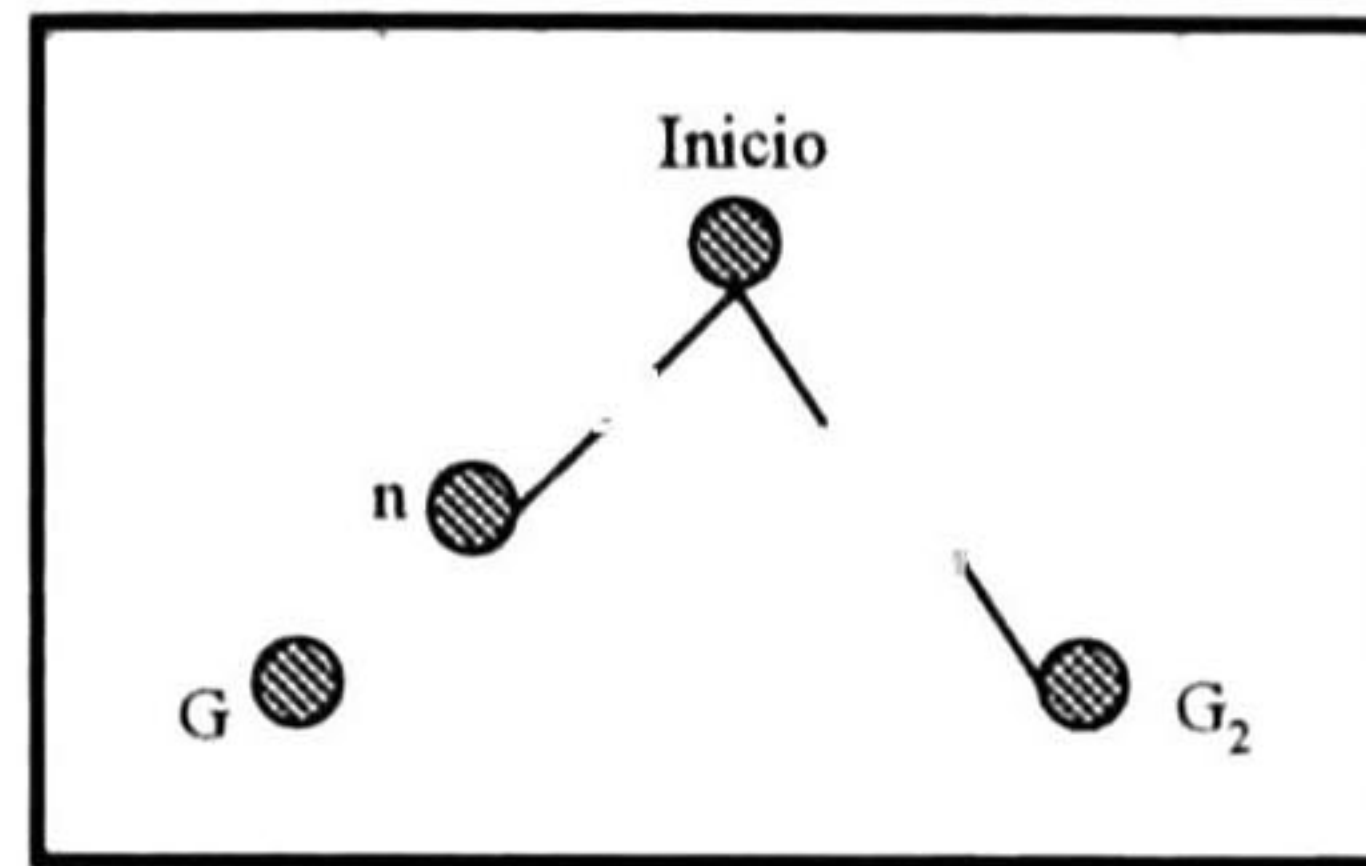


Figura 3.14: Obtención de un estado solución subóptimo.

con la búsqueda (ver la figura 3.14). Para probar la optimalidad de A^* mostraremos que esto no es posible.

Si consideramos un nodo n que es un nodo en la ruta óptima hacia G , debido a que h es admisible debemos tener que:

$$f^* \geq f(n).$$

Además, si n no es escogido en la expansión que condujo a G_2 , debemos tener:

$$f(n) \geq f(G_2).$$

Combinando las expresiones anteriores, tenemos:

$$f^* \geq f(G_2).$$

Ahora bien, debido a que G_2 es un estado meta, $h(G_2)=0$; así que $f(G_2)=g(G_2)$. Entonces, con esto probamos que:

$$f^* \geq g(G_2).$$

Esto contradice la hipótesis de que G_2 es subóptimo, entonces A^* nunca selecciona una meta subóptima por expansión. Así que, debido a que este algoritmo solo regresa una solución después de seleccionarla por expansión, A^* es un algoritmo óptimo.

PRUEBA DE QUE EL ALGORITMO ES COMPLETO.

Como se mencionó anteriormente, A^* al expandir los nodos del grafo incrementa f , eventualmente la expansión conducirá a alcanzar la meta. Esto será verdadero a menos que exista un número infinito de nodos con $f(n) < f^*$.

La única manera de que exista un número infinito de nodos es por uno de los siguientes casos:

- Existe un nodo con un factor de ramificación infinito.
- Existe una trayectoria con un costo finito pero con un infinito número de nodos.

Entonces, la descripción correcta es que A^* es completo en *grafos localmente finitos*² En la representación del mapa por medio del grafo de conectividad formado en la descomposición

²Los grafos localmente finitos son aquellos con un factor de ramificación finito.

de celdas propuesto, siempre obtenemos grafos localmente finitos, por lo que el algoritmo A^* resulta completo.

COMPLEJIDAD DE A^* .

La complejidad del algoritmo A^* puede ser una limitante para ciertos problemas. El número de nodos dentro del contorno de la búsqueda puede ser exponencial cuando se tiene un número muy grande de nodos en el espacio de estados, si la función heurística no es adecuada. El crecimiento exponencial ocurre a menos que el crecimiento en el error de la función heurística no sea mayor que el logaritmo del costo actual de la ruta. Es decir:

$$|h(n) - h^*(n)| \leq O(\log h^*(n)) ; \text{ donde } h^*(n) \text{ es el costo real del nodo } n \text{ a la meta.}$$

El crecimiento exponencial representa un problema para determinar la ruta, ya que en la búsqueda se puede consumir un tiempo también exponencial. Para evitar estos problemas se han desarrollado estrategias para reforzar el algoritmo A^* y reducir los requerimientos de espacio, entre estas técnicas están la búsqueda A^* de Profundidad Iterativa (IDA^*) y la búsqueda A^* Simplificada de Memoria Limitada (SMA^*).

Una vez que se determina la ruta óptima se debe generar la trayectoria que seguirá el vehículo durante su recorrido desde la configuración inicial a la configuración final. En este capítulo se mostró el método empleado para determinar la ruta óptima a partir de una representación del mapa por medio de un grafo. El algoritmo implementado se resume en el Apéndice 1. En el capítulo 4 se especifica la manera de generar la trayectoria para el vehículo y el control para el seguimiento de la misma. En el capítulo 5 se presentan las tareas reactivas del AGV para evadir obstáculos que se presentan en el recorrido de su trayectoria.

Capítulo 4

GENERACIÓN Y SEGUIMIENTO DE TRAYECTORIAS.

En el capítulo anterior se establecieron los criterios para generar el grafo de conectividad que representa el mapa de un espacio libre para un AGV. Además, se determinó el método de búsqueda de la ruta en el grafo de conectividad, para determinar aquella que sea óptima. En los capítulos anteriores mencionamos también que la ruta generada en la búsqueda es una secuencia de nodos en el grafo de conectividad, que representan ciertas posiciones en C_f . En el capítulo anterior también se explicó que una trayectoria para el vehículo podría ser generada por los nodos que forman la ruta óptima encontrada y sus arcos respectivos. Sin embargo, una trayectoria generada de esta manera puede resultar inconveniente para un vehículo en ciertos casos en que el grafo forme ángulos muy cerrados entre dos arcos consecutivos en la ruta. Por esta razón es conveniente suavizar esta primera aproximación a la trayectoria del vehículo para obtener curvas continuas más adecuadas. Es cierto que buscamos en el grafo de conectividad una ruta tal que fuera lo más cercana posible a la trayectoria final del vehículo, pero sin duda habrá diferencia entre la trayectoria sin suavizar que produciría el grafo de conectividad y la trayectoria suavizada por medio de B-splines, que se analizará en este capítulo.

En este capítulo también mostraremos el modelo del sistema empleado en el simulador. El modelo básicamente consiste en un modelo matemático del vehículo que se implementó con el fin de analizar el efecto de los algoritmos de control en el seguimiento de la trayectoria, que se analiza posteriormente.

4.1 B-SPLINES.

La función que tienen las B-splines en el sistema de planeación de movimiento del AGV es el de generar la trayectoria suavizada que seguirá el vehículo para llegar de la configuración

inicial a la configuración final. La generación de la trayectoria es un proceso que se hace mientras que el vehículo está en movimiento por medio de las B-splines. Estas son función del tiempo e indicarán al vehículo una secuencia continua de configuraciones que deberá seguir.

Las B-splines y otros términos útiles para nuestro análisis se definen a continuación [26, 27, 17]:

Definición 43 Una B-spline es una curva paramétrica que une dos puntos en el plano, la cual es generada analíticamente al proponer el grado de un polinomio que pondera los puntos que definen a la B-spline, que son llamados puntos de control.

Definición 44 Los puntos de control unidos por segmentos de línea recta forman el llamado polígono de control.

Definición 45 La continuidad de la B-spline se da por la unión de dos segmentos consecutivos de la curva en el punto llamado punto nudo.

Definición 46 Si las derivadas de una curva $P(t)$, $a \leq t \leq b$ de orden r existen y son continuas, y si $\dot{\vec{P}} \neq 0$ en todo el intervalo, entonces la curva es de clase C^r

Definición 47 Sea $P(t)$ una curva parametrizada, el radio de curvatura ρ en un punto t es el radio de un círculo que pasa por t y que coincide con la curva en una vecindad de t . La curvatura es el inverso de ρ , $\kappa = \frac{1}{\rho}$. En términos de t la curvatura está dada por:

$$\kappa(t) = \frac{|\dot{\vec{P}}(t) \times \ddot{\vec{P}}(t)|}{|\dot{\vec{P}}(t)|^3} \quad (4.1)$$

y el vector de curvatura está dado por: $\ddot{\vec{P}}(s) = \kappa \vec{n}$,
donde \vec{n} es el vector unitario que apunta hacia el centro de curvatura.

La curva B-spline se forma al proponer una combinación lineal en términos de funciones polinomiales. Se pueden utilizar funciones polinomiales cuadráticas o cúbicas, que son las más utilizadas en la práctica. Las funciones polinomiales que utilizaremos son polinomios cuadráticos, por lo que las B-splines generadas reciben el nombre de *B-splines cuadráticas*. Para las B-splines cuadráticas se tiene la combinación lineal vectorial siguiente:

$$\vec{P}_i(t) = X_0(t) \vec{Q}_{i-1} + X_1(t) \vec{Q}_i + X_2(t) \vec{Q}_{i+1}, \quad (4.2)$$

donde $i=1, 2, \dots, n-2$.

Cada término $\vec{P}_i(t)$ representa un segmento de la curva generado al variar t . Las funciones $X_i(t)$ son polinomios en términos del parámetro t , donde $0 \leq t \leq 1$. Para la construcción de las curvas B-spline cuadráticas es necesario satisfacer las siguientes condiciones [25]:

1. La curva constituida por dos segmentos consecutivos $\vec{P}_i(t)$ y $\vec{P}_{i+1}(t)$ es continua, es decir:

$$\vec{P}_i(t = 1) = \vec{P}_{i+1}(t = 0) \quad (4.3)$$

Estos puntos de la curva son los puntos de unión que aseguran la continuidad de la curva y son los llamados *puntos nudo*.

2. La primera derivada de dos segmentos contiguos son continuas.

$$\dot{\vec{P}}_i(t = 1) = \dot{\vec{P}}_{i+1}(t = 0) \quad (4.4)$$

3. La propiedad de invarianza bajo una transformación de coordenadas se establece por la condición de Cauchy como sigue (referirse a [25] para ver detalles):

$$X_0(t) + X_1(t) + X_2(t) = 1 \quad (4.5)$$

Las tres condiciones anteriores definen las llamadas *funciones básicas*. De la primera condición tenemos:

$$X_0(t = 1)Q_{i-1} + X_1(t = 1)Q_i + X_2(t = 1)Q_{i+1} = X_0(t = 0)Q_i + X_1(t = 0)Q_{i+1} + X_2(t = 0)Q_{i+2}.$$

para todo Q_j ($j=i-1, \dots, i+2$).

Igualando coeficientes tenemos:

$$\begin{aligned} X_0(t = 0) &= X_1(t = 1) \\ X_1(t = 0) &= X_2(t = 1) \\ X_0(t = 0) &= X_2(t = 1) = 0 \end{aligned} \quad (4.6)$$

Y de igual forma para $\dot{X}_i(t)$, se tiene:

$$\begin{aligned}\dot{X}_0(t=0) &= \dot{X}_1(t=1) \\ \dot{X}_1(t=0) &= \dot{X}_2(t=1) \\ \dot{X}_0(t=0) &= \dot{X}_2(t=1) = 0\end{aligned}$$

Proponiendo los términos $X_i(t)$ como polinomios de segundo grado tenemos:

$$\begin{aligned}X_0(t) &= X_2(1-t) \\ X_1(t) &= a_1t^2 + b_1t + c_1 \\ X_2(t) &= a_2t^2 + b_2t + c_2\end{aligned}\tag{4.7}$$

De las ecuaciones 4.6 tenemos:

$$\begin{aligned}X_2(t=0) = 0 &\implies c_2 = 0 \\ \dot{X}_2(t=0) = 0 &\implies b_2 = 0\end{aligned}$$

$$X_1(t=0) = X_2(t=1) \implies c_1 = a_2\tag{4.8}$$

$$\dot{X}_1(t=0) = \dot{X}_2(t=1) \implies 2a_1t + b_1 \Big|_{t=0} = 2a_2t \Big|_{t=1} \implies b_1 = a_2\tag{4.9}$$

De la ecuación 4.5 tenemos :

$$\begin{aligned}X_0(t) + X_1(t) + X_2(t) &= 1 \\ X_2(1-t) + a_1t^2 + b_1t + c_1 + a_2t^2 &= 1 \\ a_2(1-t)^2 + a_1t^2 + 2a_2t + a_2 + a_2t^2 &= 1 \implies a_2 = \frac{1}{2}, a_1 = -1\end{aligned}$$

Sustituyendo en las ecuaciones 4.9 y 4.8:

$$b_1 = 1, c_1 = \frac{1}{2}.$$

Con los coeficientes anteriores y sustituyendo en las ecuaciones 4.7 obtenemos las siguientes funciones:

$$\begin{aligned}X_0(t) &= \frac{1}{2}(1-t)^2 \\ X_1(t) &= -t^2 + t + \frac{1}{2} \\ X_2(t) &= \frac{1}{2}t^2\end{aligned}$$

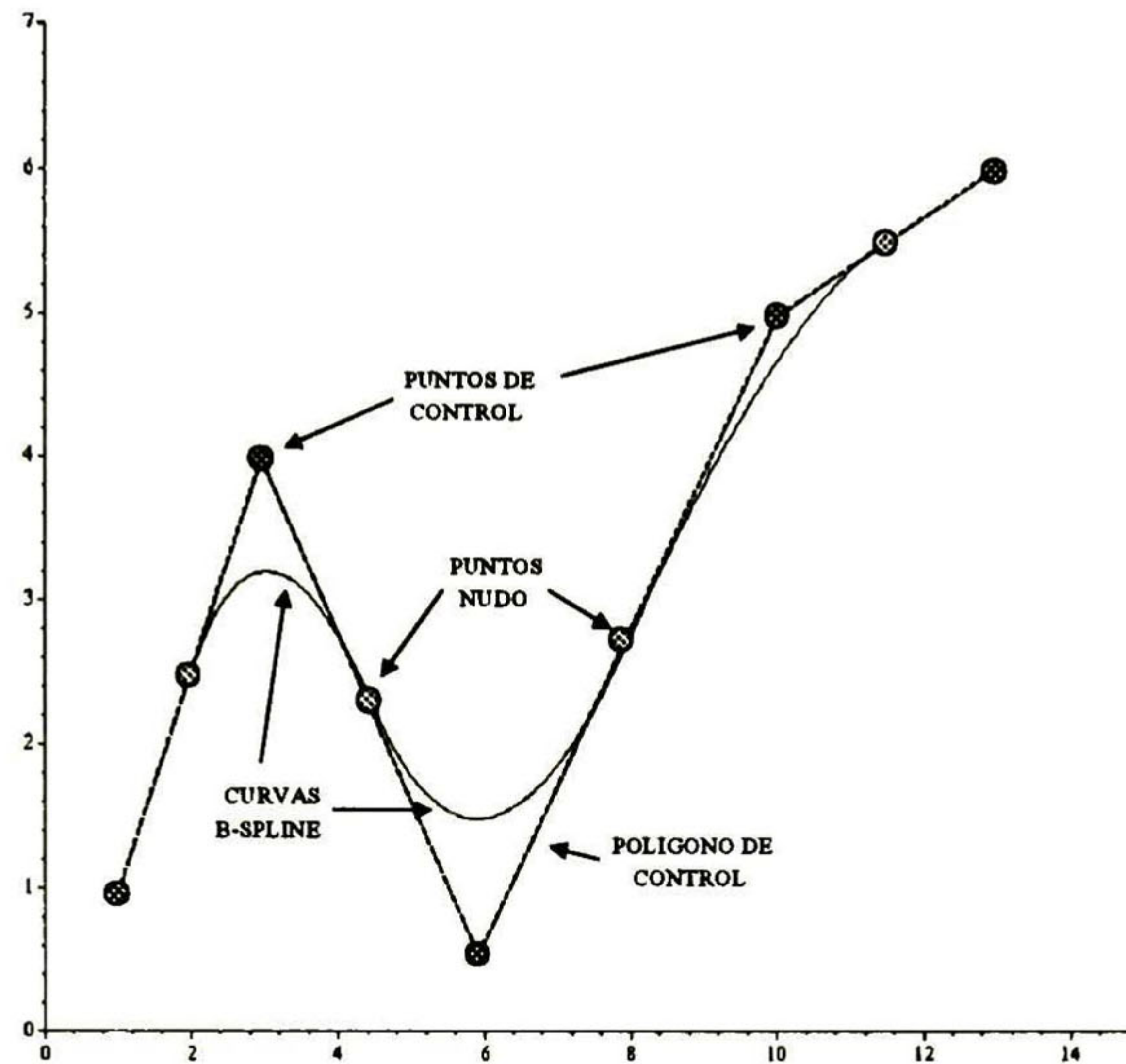


Figura 4.1: En la figura se muestra un ejemplo de curvas B-spline cuadráticas.

A estas ecuaciones se les llama *funciones básicas uniformes* de orden $M=3$, por utilizar una combinación lineal de tres puntos consecutivos, y de grado dos por utilizar polinomios cuadráticos. Cuando los términos $P_i(t)$ se generan a partir de las funciones básicas uniformes se denominan *B-splines*. La figura 4.1 muestra un ejemplo de B-splines cuadráticas.

En la sección siguiente se presentan las características de las B-splines que las hacen adecuadas para generar la trayectoria del AGV en el simulador.

4.2 PROPIEDADES DE LAS B-SPLINES.

Las B-splines son curvas paramétricas que poseen algunas características especiales que las hacen apropiadas para la tarea de generación de la trayectoria del AGV. Las características más importantes en la implementación se describen a continuación; otras propiedades de las B-splines se describen en [25]:

1. La curva formada por las B-splines es clase C^1 . Dos segmentos consecutivos $\vec{P}_i(t)$ y $\vec{P}_{i+1}(t)$ son continuos por la condición 1, ecuación 4.3. La derivada $\dot{\vec{P}}_i(t) \forall i$ es continua por la condición 2, ecuación 4.4. Además, se cumple la condición $\dot{\vec{P}}_i(t) \neq 0 \forall i$ en un intervalo $0 \leq t \leq 1$; por lo que la B-spline cuadrática es clase C^1 .

Esta propiedad nos garantiza que la trayectoria generada es una secuencia continua de configuraciones que seguirá el vehículo, y al mismo tiempo determina que el vehículo tendrá una trayectoria de velocidad continua.

2. *Convexidad geométrica con respecto al polígono de control.* Cada segmento B-spline se forma por sus tres vectores de control respectivos que constituyen a su vez un polígono de control. Esta propiedad indica que cada segmento B-spline está contenido en la convexidad geométrica de su polígono de control (ver la figura 4.2). (Más detalles en [25]).

Esta característica podría representar dificultades en la trayectoria generada para el vehículo, ya que la trayectoria respectiva podría causar colisión del vehículo con algún obstáculo como se muestra en la figura 4.3, donde la propiedad de convexidad puede ocasionar colisiones en la parte superior del obstáculo A y en la esquina inferior izquierda de B. Sin embargo, este problema se resolverá con las tareas reactivas de evasión de obstáculos, que se presenta en el capítulo 5.

3. *Variabilidad.* Las B-splines cuadráticas se definen localmente por tres puntos de control, que corresponden a tres puntos consecutivos del polígono de control. El cambio de uno de los puntos de control \vec{Q}_i , causará solamente variaciones locales sobre los tres segmentos implicados sin afectar el resto. (Ver la figura 4.4).

La característica de variabilidad es importante para la tarea reactiva de evasión de obstáculos, que se realiza al modificar la trayectoria localmente. Este tema se verá con mayor detalle en el capítulo 5.

4. *Localización de los puntos nudo.* Los puntos nudo de las B-splines se localizan en los puntos medios de los segmentos del polígono de control. De la fórmula 4.2 se verifica fácilmente que:

$$\begin{aligned} P_i(t=0) &= \frac{1}{2}(\vec{Q}_{i-1} + \vec{Q}_i) & \forall i & \text{ además,} \\ P_i(t=1) &= \frac{1}{2}(\vec{Q}_i + \vec{Q}_{i+1}) & \forall i. \end{aligned}$$

Esta característica permite predecir que la trayectoria descrita por las B-splines cruzará todos los puntos medios de los segmentos del polígono de control, los cuales corresponden a los arcos formados en el grafo de conectividad.

5. *Segmentos tangentes.* A diferencia de las B-splines cúbicas, en los puntos nudo de las B-splines cuadráticas la tangente son los segmentos del polígono de control. Esto se puede ver al derivar la fórmula 4.2 y evaluar en $t=0 \forall i$, obteniendo:

$$\dot{\vec{P}}_i(t=0) = \vec{Q}_i - \vec{Q}_{i-1},$$

que corresponde al i -ésimo segmento del polígono de control.

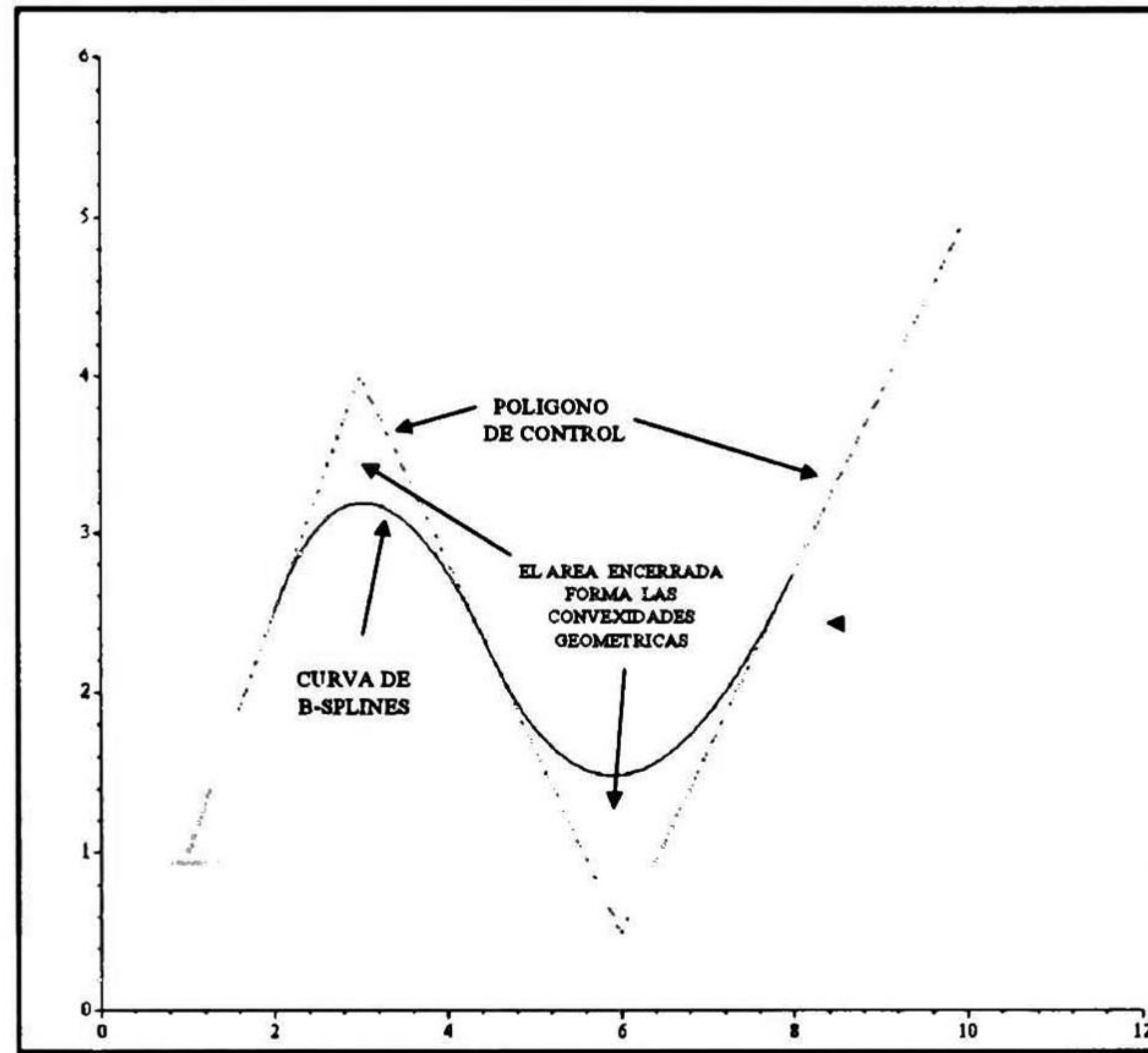


Figura 4.2: Propiedad de convexidad geométrica.

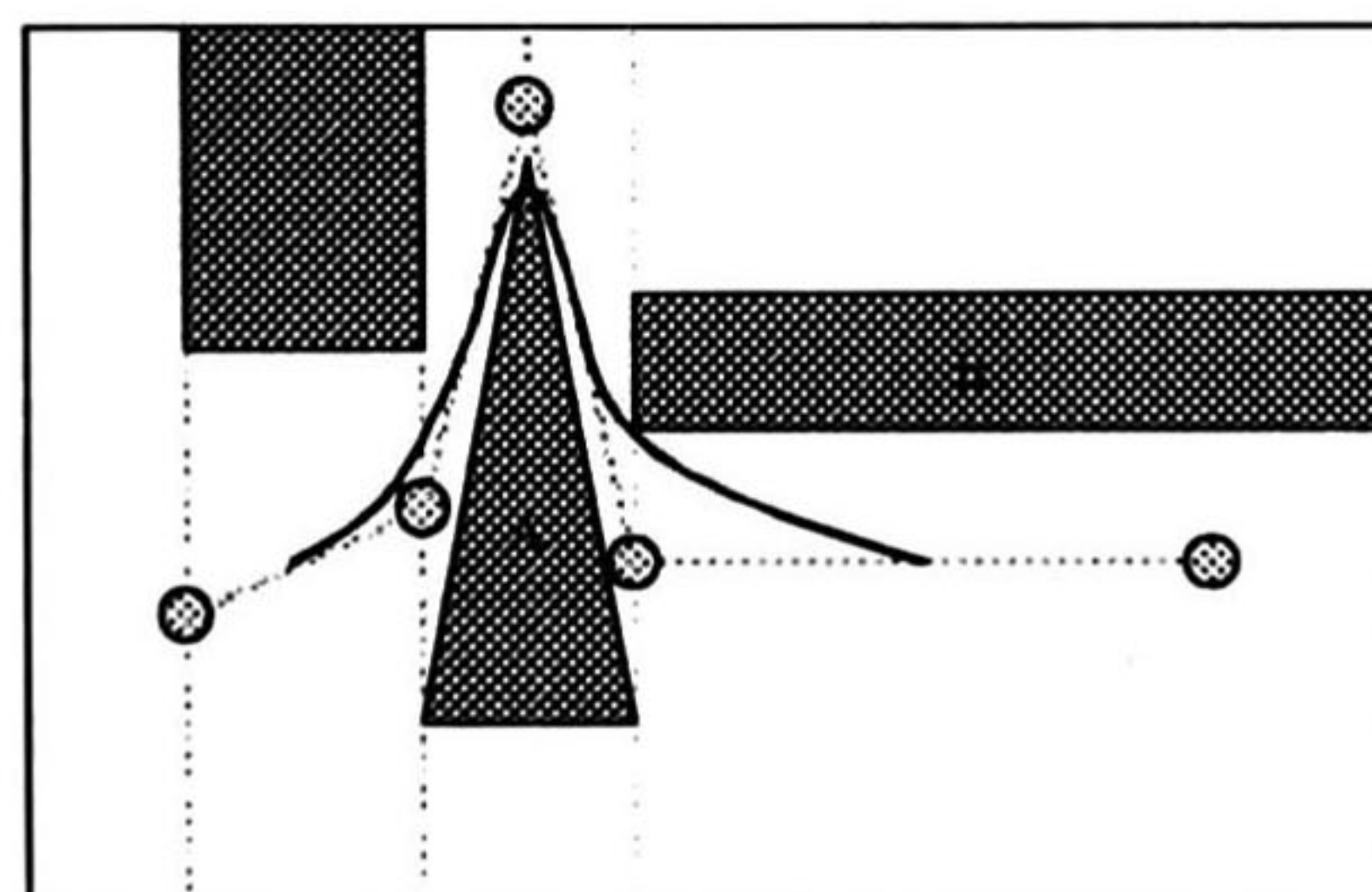


Figura 4.3: Colisión por convexidad geométrica de las B-splines.

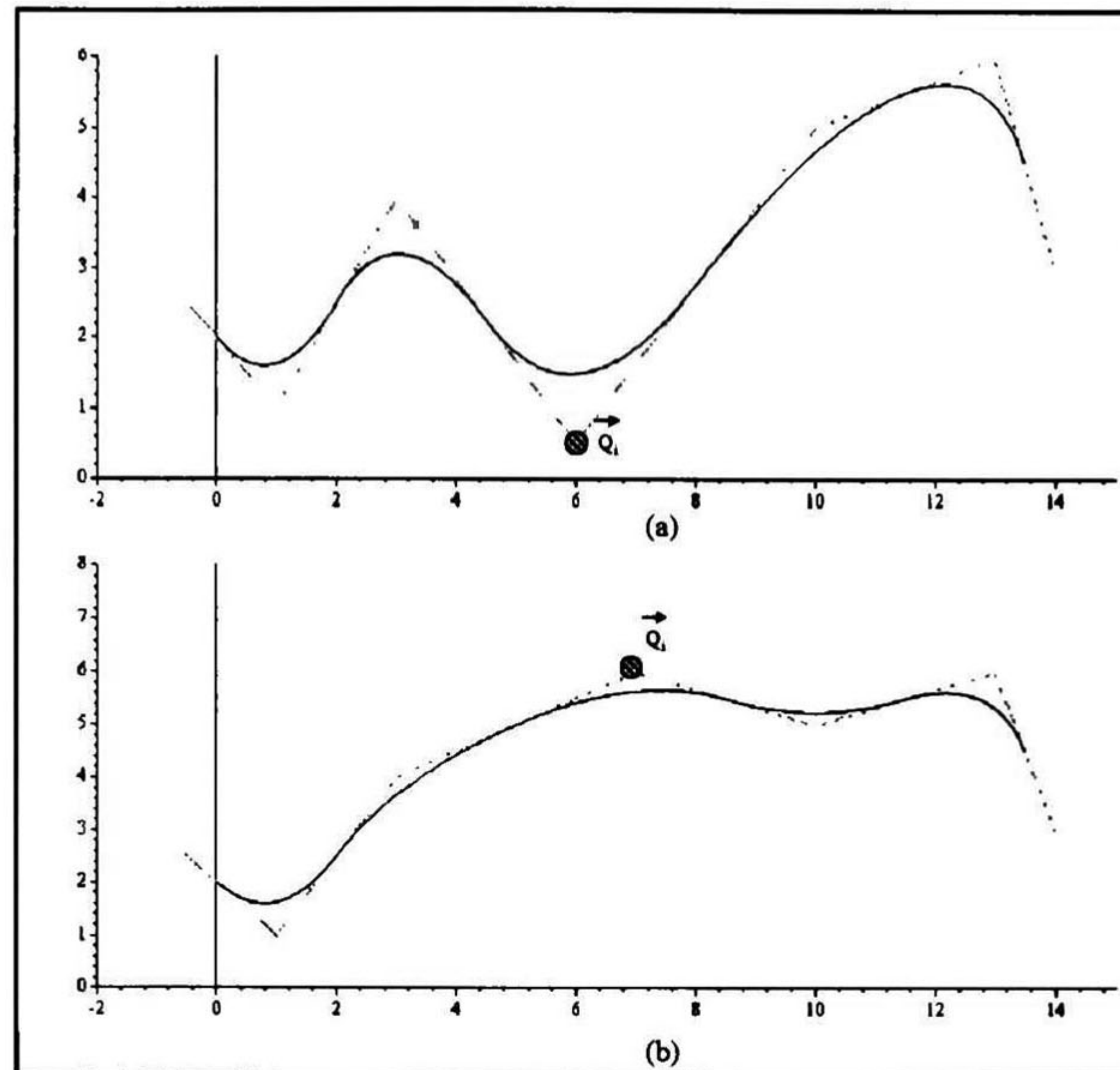


Figura 4.4: Modificación de las B-splines al variar el punto de control \vec{Q}_i .

Además de las características de las B-splines cuadráticas descritas, las B-splines contienen información implícita en su representación que nos permite generar la trayectoria para el AGV; en las siguientes ecuaciones se presenta la información que se determina a partir de las B-splines. La ecuación 4.2 se puede reescribir como sigue:

$$\begin{aligned} x_i(t) &= b_0(t)p_{i-1} + b_1(t)p_i + b_2(t)p_{i+1}, \\ y_i(t) &= b_0(t)q_{i-1} + q_1(t)p_i + q_2(t)p_{i+1}, \end{aligned} \quad (4.10)$$

donde las funciones $b_i(t)$, con $i=0, 1, 2$, son las funciones básicas.

Sea φ el ángulo de la tangente a la curva Γ en un punto $Z=(x(t),y(t))$, donde las funciones $x(t)$ y $y(t)$ determinan las coordenadas de Z para un cierto valor de t , definido como el tiempo. El punto Z se puede representar en Γ con un vector de posición $\vec{P}(t)$ y una longitud l que se mide desde un extremo de la curva Γ (ver la figura 4.5). De esta manera el desplazamiento en la curva Γ se puede medir en términos de las características de la B-spline [25].

Bajo los términos de las B-splines se determinan las expresiones siguientes (ver el desarrollo en [25]):

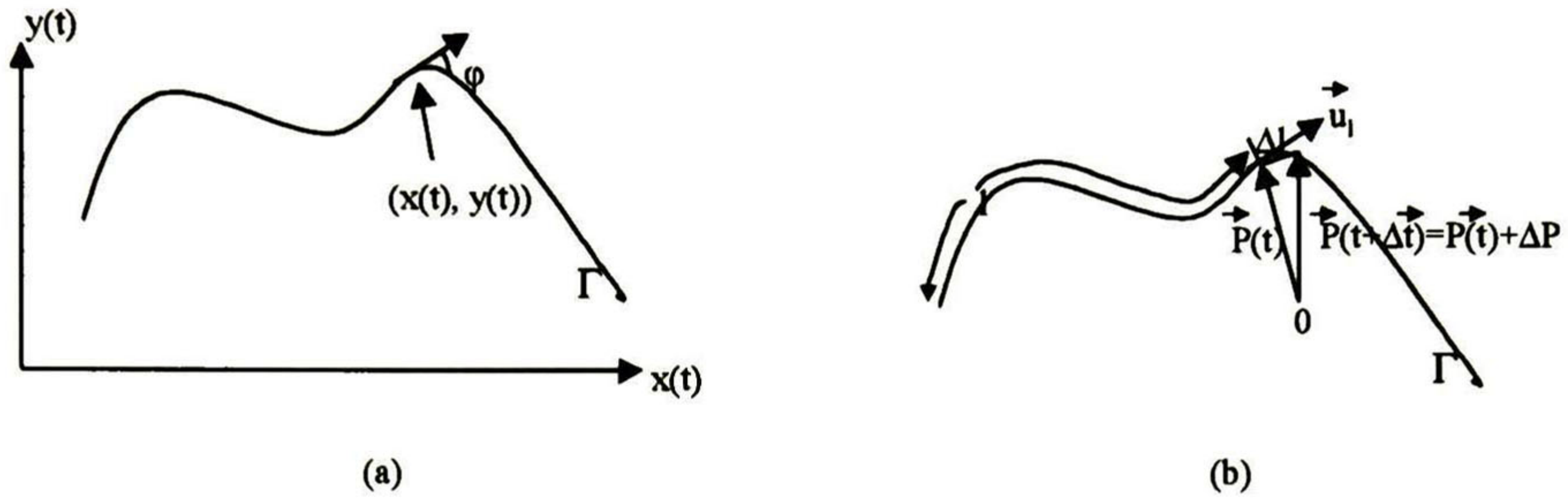


Figura 4.5: La figura (a) muestra la interpretación del parámetro t . En (b) se determina un punto sobre Γ en términos de $P(t)$, l y t .

$$\vec{P}(t) = x(t)\hat{i} + y(t)\hat{j} \tag{4.11}$$

donde los vectores \hat{i} y \hat{j} representan los vectores unitarios en dirección positiva de los ejes x y y respectivamente.

Por lo tanto

$$\begin{aligned} \frac{d\vec{P}(t)}{dt} &= \frac{dx(t)}{dt}\hat{i} + \frac{dy(t)}{dt}\hat{j}, \text{ o bien} \\ \frac{d\vec{P}(t)}{dt} &= v_x\hat{i} + v_y\hat{j} \end{aligned}$$

donde

$$\tan \varphi = \frac{\frac{dy(t)}{dt}}{\frac{dx(t)}{dt}}$$

y

$$\varphi = \arctan\left(\frac{\dot{y}}{\dot{x}}\right) \tag{4.12}$$

Por el *teorema de los valores extremos*¹ de las funciones [27], tenemos un máximo y un mínimo para la función de velocidad descrita por las B-splines y estos se determinan de su información implícita. En las B-splines cuadráticas los segmentos $P_i(t)$ que forman la curva de la trayectoria tienen uno o dos valores críticos² donde la velocidad es máxima y estos

¹El teorema de los valores extremos establece que una función f continua en un intervalo cerrado $[a, b]$ siempre tiene un máximo absoluto y un mínimo absoluto en el intervalo.

²Un *valor crítico* de una función f es un número c en su dominio para el cual $f'(t)=0$ o $f'(c)$ no existe [Zill,1985].

son los puntos nudo, en $P_i(t = 0)$ y $P_i(t = 1)$. Además, si existe un mínimo en la función de velocidad, descrita por la curva B-spline $P_i(t)$, dentro del intervalo $0 < t < 1$, el punto crítico del mínimo de la velocidad se localiza en $P_i(t = t_{\min})$, y el parámetro t_{\min} está dado por:

$$t_{\min} = \frac{-(ab + cd)}{a^2 + c^2} \quad (4.13)$$

donde:

$$\begin{aligned} a &= p_{i-1} - 2p_i + p_{i+1} \\ b &= -p_{i-1} + p_i \\ c &= q_{i-1} - 2q_i + q_{i+1} \\ d &= -q_{i-1} + q_i \end{aligned} \quad (4.14)$$

De no existir un mínimo en el intervalo $0 < t < 1$, el mínimo de la función de velocidad se encuentra en alguno de los puntos nudo.

Para probar esta propiedad de las B-splines, se determina la función de velocidad a partir de las ecuaciones 4.10 y 4.11, obteniendo:

$$\begin{aligned} \frac{d\vec{P}(t)}{dt} &= [(t-1)p_{i-1} + (-2t+1)p_i + (t)p_{i+1}]\hat{i} + [(t-1)q_{i-1} + (-2t+1)q_i + (t)q_{i+1}]\hat{j} \\ &= (at+b)\hat{i} + (ct+d)\hat{j} \end{aligned}$$

donde a, b, c y d son los términos de las ecuaciones 4.14.

Para determinar la magnitud de la velocidad tenemos:

$$v_m = \left| \frac{d\vec{P}(t)}{dt} \right| = \sqrt{(at+b)^2 + (ct+d)^2} \quad (4.15)$$

Por el teorema 4.3³ de [27], identificamos los extremos absolutos⁴ en $t = 0$, $t = 1$ y en algún valor crítico que se presente para v_m en el intervalo $0 < t < 1$. El valor crítico se determina derivando la ecuación 4.15 e igualando a cero, obteniendo:

³Si f es continua en un intervalo cerrado $[a, b]$, entonces un extremo absoluto ocurre en un punto frontera del intervalo o en un valor crítico en el intervalo abierto (a, b) .

⁴Son los valores máximo y mínimo de la función en un intervalo.

$$\dot{v}_m = \frac{(a^2 + c^2)t + (ab + cd)}{\sqrt{(at + b)^2 + (ct + d)^2}} = 0 \quad (4.16)$$

De la ecuación 4.16 se desprende el valor crítico dado en la fórmula 4.13. Este valor crítico, si está dentro del intervalo $0 < t < 1$, es un mínimo ya que cumple con el teorema 4.8⁵ de [27] al tener:

$$\begin{aligned} \dot{v}_m |_{t=0^+} &= \frac{ab+cd}{\sqrt{b^2+d^2}} \\ \dot{v}_m |_{t=1^-} &= \frac{(a^2+c^2)+(ab+cd)}{\sqrt{(a+b)^2+(c+d)^2}} \end{aligned}$$

Los denominadores son ambos positivos, y como $0 < t_{\min} < 1$, entonces se cumplen:

$$\begin{aligned} ab + cd < 0 \quad \text{si} \quad 0 < t_{\min}, \quad \text{entonces} \quad \dot{v} |_{t=0^+} < 0 \quad \text{y,} \\ a^2 + c^2 > -(ab + cd) \quad \text{si} \quad t_{\min} < 1, \quad \text{entonces} \quad \dot{v} |_{t=1^-} > 0. \end{aligned}$$

Así que el mínimo se encuentra en t_{\min} y el máximo está en uno o ambos puntos nudo. Si el valor crítico encontrado con la fórmula 4.13 está fuera del intervalo $0 < t < 1$, entonces el mínimo de la función de velocidad en $P_i(t)$ se encuentra en alguno de los puntos nudo y el máximo en el otro. En las figuras 4.6, 4.7 y 4.8 se presentan varias gráficas de velocidad para diferentes configuraciones de puntos de control.

El radio de curvatura ρ de la curva descrita por las B-splines es el inverso de la curvatura κ dado por la fórmula 4.1. Por la construcción de las B-splines cuadráticas, la ecuación que define el radio de curvatura se reduce a la expresión:

$$\rho = \frac{|\dot{\vec{P}}(t)|^3}{(q_0 - 2q_1 + q_2)(-p_0 + p_1) - (p_0 - 2p_1 + p_2)(-q_0 + q_1)}$$

El radio de curvatura de la B-spline está relacionado con la velocidad, de manera que el mínimo y el máximo radio de curvatura ocurren en el mismo tiempo t donde ocurren el mínimo y máximo de la velocidad respectivamente. Esto se observa en las gráficas de las figuras 4.6, 4.7 y 4.8.

Como el máximo de la velocidad de un segmento se da en $t = 0$ o $t = 1$ la velocidad máxima está dada por: $\dot{\vec{P}}_i(t = 0) = \vec{Q}_i - \vec{Q}_{i-1}$ o $\dot{\vec{P}}_i(t = 1) = \vec{Q}_{i+1} - \vec{Q}_i$. Este hecho puede ser útil para controlar en cierta forma la velocidad del vehículo. Supongamos que

⁵El teorema establece el criterio de la primera derivada para extremos relativos: Sea f continua en $[a, b]$ y diferenciable en (a, b) , excepto posiblemente en el valor crítico c . Si $f'(x) < 0$ para $a < x < c$ y $f'(x) > 0$ para $c < x < b$, entonces $f(c)$ es un mínimo relativo.

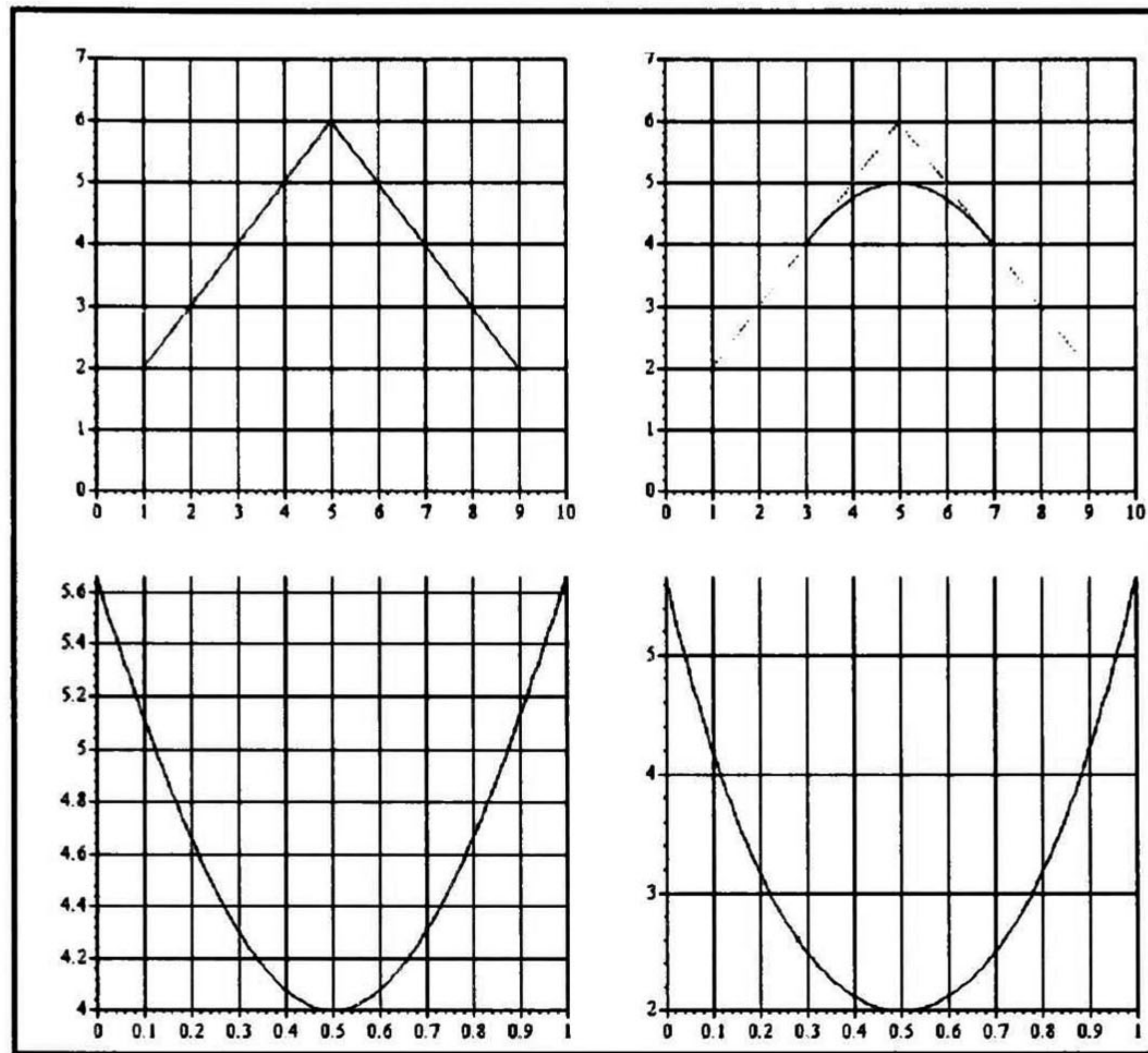


Figura 4.6: Puntos máximos de velocidad en ambos puntos nudo y el punto mínimo dentro de $0 < t < 1$.

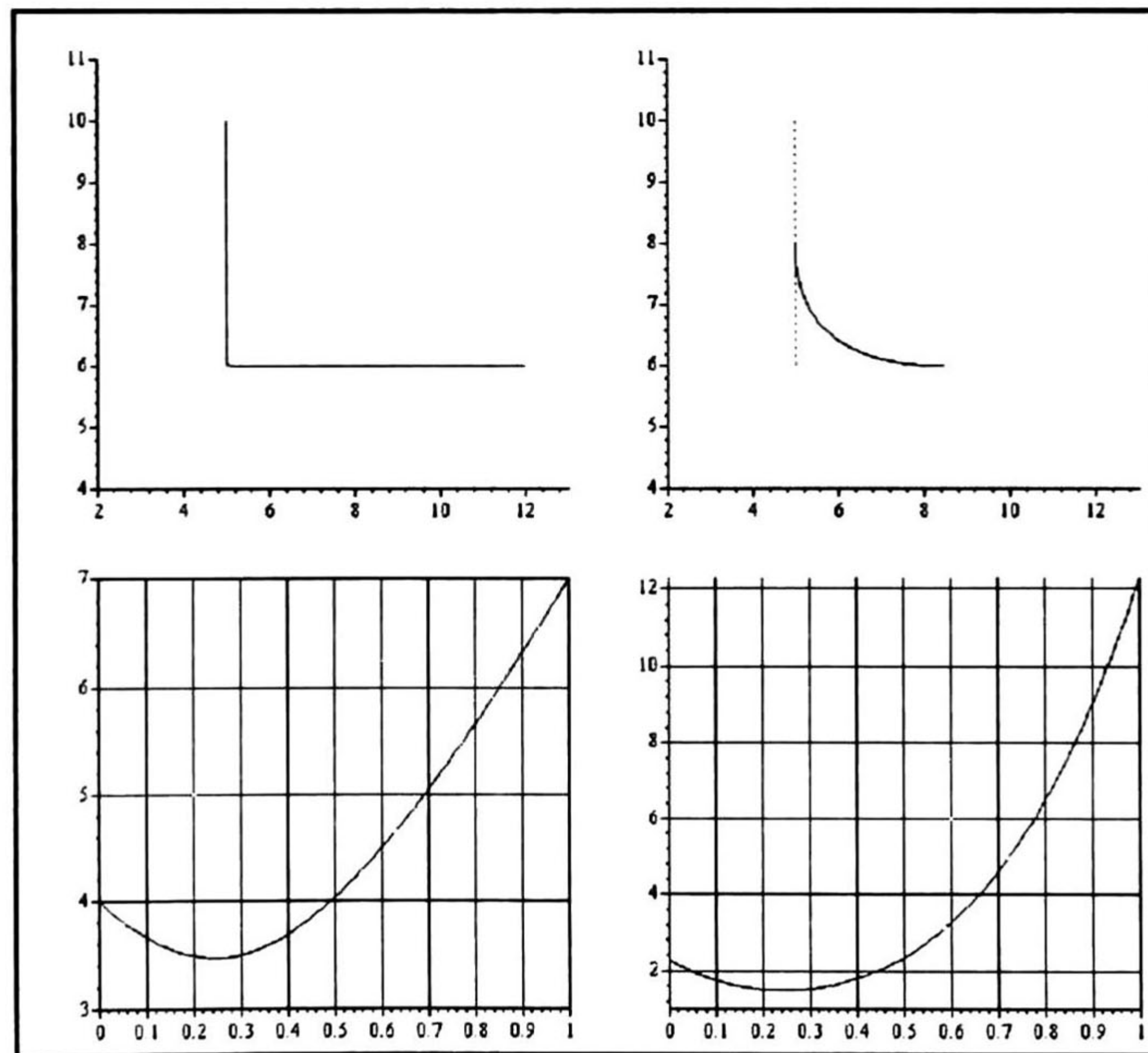


Figura 4.7: Punto máximo de velocidad en el punto nudo $t = 1$, y punto mínimo dentro del intervalo $0 < t < 1$.

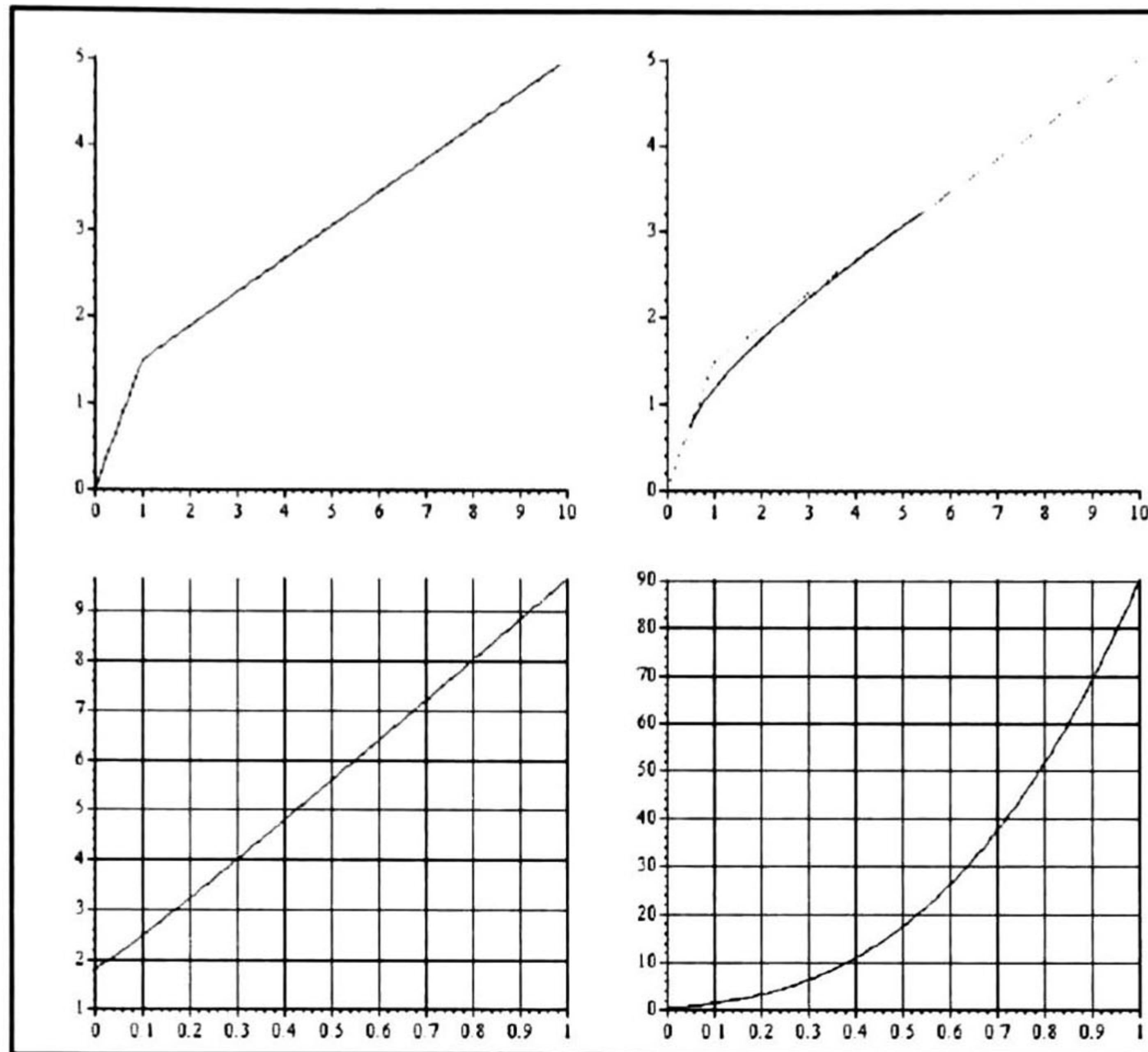


Figura 4.8: Punto de velocidad máxima en el punto nudo $t = 1$, y mínima en el punto nudo $t = 0$.

tenemos dos nodos consecutivos, W y Z , en la ruta definida en el grafo de conectividad, cuya separación provoque que la velocidad máxima en el punto nudo entre ellos sea mayor que la especificada para el vehículo. En este caso, se podrían incluir *nodos auxiliares de trayectoria*⁶ alineados sobre el arco que une a los nodos W y Z , de manera que se reduzca la distancia entre nodos consecutivos y reduciendo como consecuencia la velocidad máxima entre W y Z .

Es claro también, que la velocidad máxima a lo largo de la trayectoria generada por las B-splines se presenta en el punto nudo cuyos puntos de control, que forman el segmento tangente a él, tengan la mayor separación. Así que el valor máximo de velocidad en la trayectoria se podría determinar antes de iniciar la generación de las B-splines. Y de ser necesario, se podría reducir la velocidad máxima mediante nodos auxiliares de trayectoria.

Las B-splines proporcionan suficiente información para generar la trayectoria y su adecuación para el control del vehículo se muestra en la sección 4.4, después de determinarse el modelo matemático del vehículo en la sección siguiente.

⁶Son nodos que se agregan en la generación de la trayectoria para limitar la velocidad máxima pero que no son considerados como parte del grafo de conectividad.

4.3 MODELO DEL SISTEMA.

El modelo matemático implementado para el AGV del simulador corresponde a un vehículo de forma circular, representado en la figura 4.9. La forma física exterior del vehículo empleada en el modelo es muy simple, pero muchos tipos de vehículos se pueden considerar como vehículos de forma cilíndrica. Si tenemos un vehículo con forma exterior o chasis diferente al del vehículo modelado, los algoritmos de control implementados no se ven afectados significativamente, incluso en casos donde tengamos un vehículo con remolque como el mostrado en la figura 4.10; ya que para diferentes vehículos, deben modificarse solamente aquellos parámetros relacionados con las dimensiones del vehículo sin alterar la estructura de los algoritmos. En vehículos con remolque, sin embargo, se debe tener especial atención en los radios de giro para evitar posibles colisiones del remolque con obstáculos y con el propio remolcador en giros cerrados. Esto establece una restricción dinámica en el radio de curvatura realizable para el vehículo con remolque. La solución a este problema se puede dar en la etapa de preplaneación (al generar el grafo de conectividad), o bien, como una tarea reactiva. En estos casos se debe considerar:

a) *Solución en la etapa de preplaneación del movimiento.* Está claro que el radio de curvatura en una B-spline $P_i(t)$, depende del ángulo β formado por el polígono de control (entre el segmento de \vec{Q}_{i-1} y \vec{Q}_i , con el segmento de \vec{Q}_i y \vec{Q}_{i+1}). Entonces, existirá un ángulo crítico β_{\min} formado por los puntos de control \vec{Q}_{i-1} , \vec{Q}_i y \vec{Q}_{i+1} , tal que en la trayectoria descrita por $P_i(t)$ el vehículo llegue a su radio de curvatura mínimo posible ρ_{\min} . Si el ángulo β se limita para cada conjunto de tres nodos consecutivos en el grafo de conectividad de manera que para toda secuencia posible de tres puntos de control \vec{Q}_{i-1} , \vec{Q}_i y \vec{Q}_{i+1} el ángulo mínimo formado sea mayor que el ángulo β_{\min} , entonces se puede esperar que para cualquier ruta determinada en el grafo tendremos un radio de curvatura mayor a ρ_{\min} en la trayectoria del vehículo.

b) *Solución como tarea reactiva.* Si durante la trayectoria del vehículo se presenta un radio de curvatura menor a ρ_{\min} , se puede modificar la trayectoria localmente de manera que el radio de curvatura sea realizable para el vehículo. En el capítulo 5 se describe el procedimiento de modificación de la trayectoria.

En el vehículo modelado se consideran dos motores independientes de magneto permanente que activan las dos ruedas de tracción. El modelo matemático considerado en el simulador para el vehículo está dado por las ecuaciones siguientes (ver desarrollo en [25]):

$$\begin{aligned} \dot{\omega}_{m1} \left(\frac{R_e J_m}{K_m} + \frac{R_e R^2 \gamma_1 m}{K_m} \right) + \omega_{m1} \left(\frac{R_e B_m}{K_m} + K_b \right) &= V_1 \\ \dot{\omega}_{m2} \left(\frac{R_e J_m}{K_m} + \frac{R_e R^2 \gamma_2 m}{K_m} \right) + \omega_{m2} \left(\frac{R_e B_m}{K_m} + K_b \right) &= V_2 \end{aligned} \quad (4.17)$$

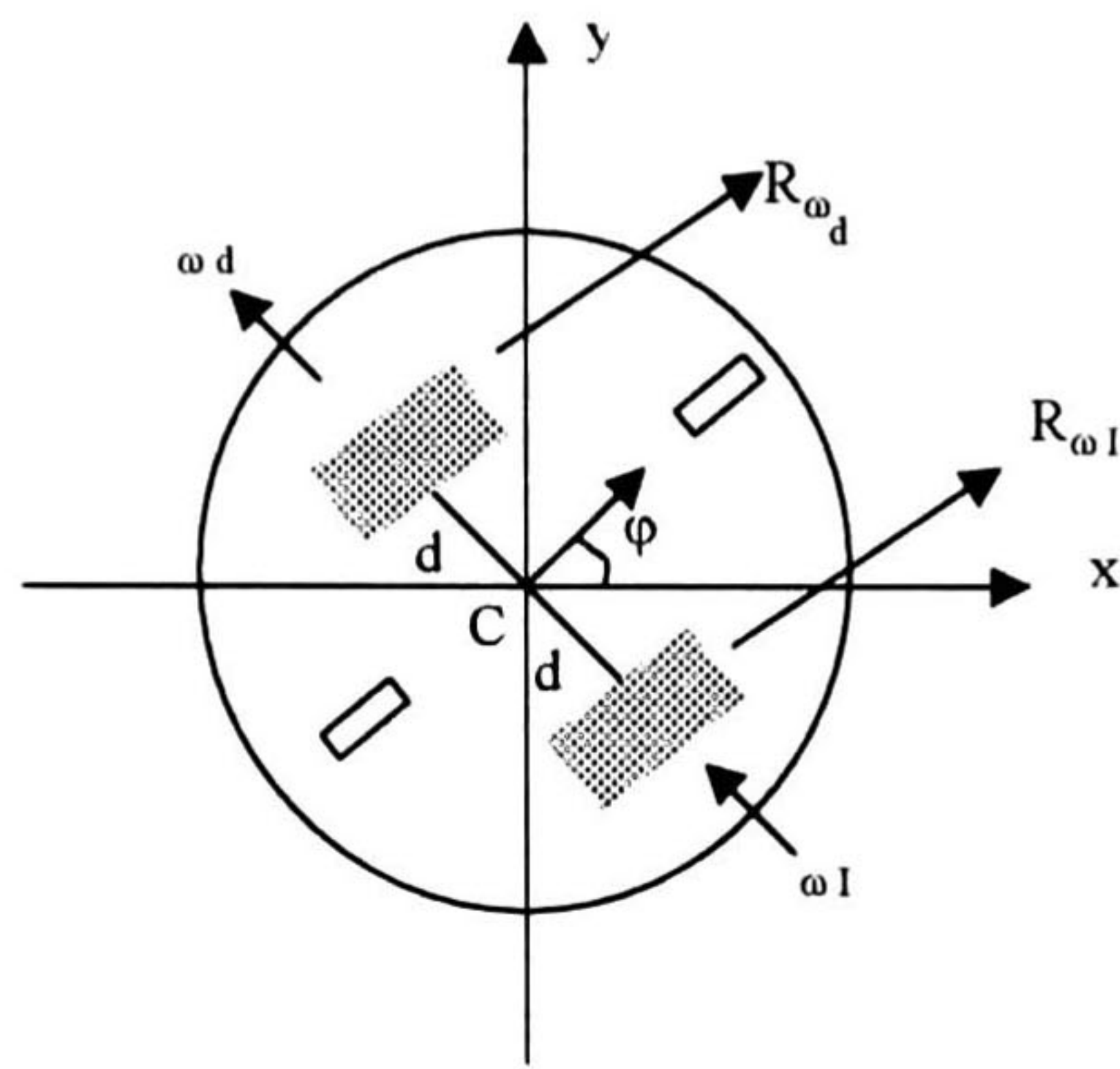


Figura 4.9: Vehículo modelado en el programa simulador.

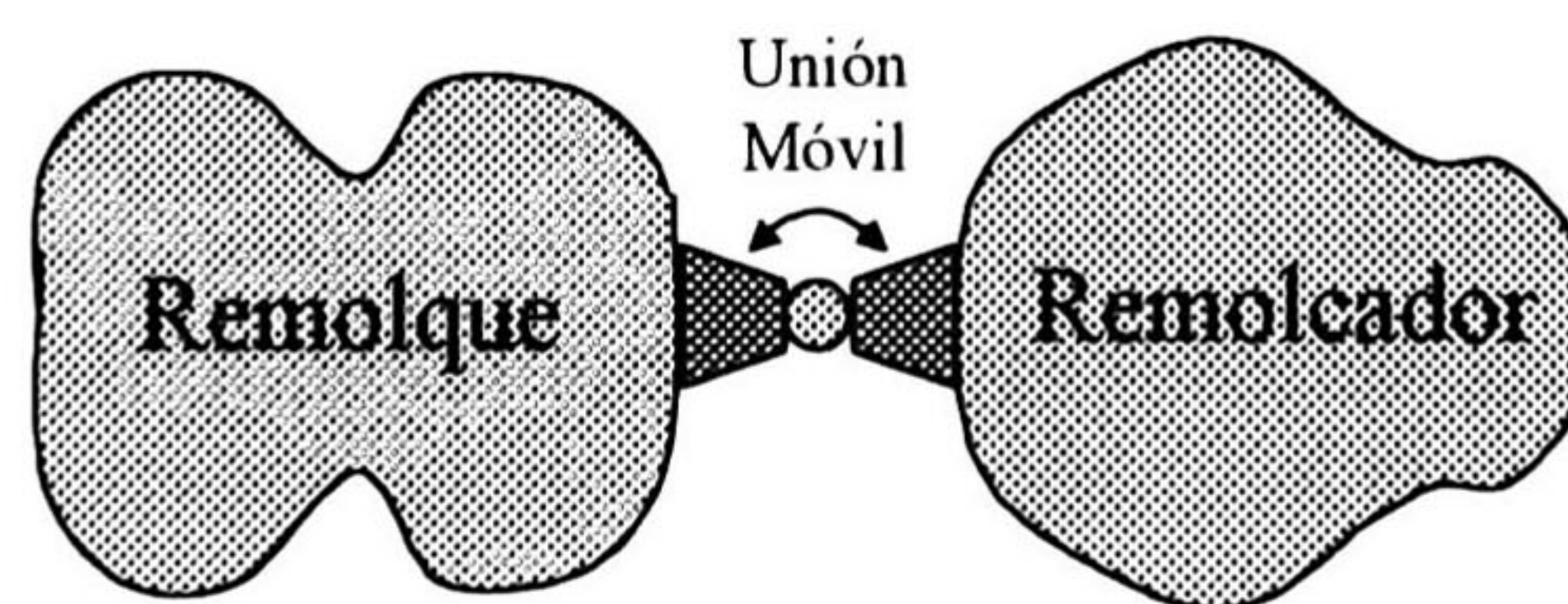


Figura 4.10: Vehículo con remolque.

donde:

- ω_m es la velocidad angular del rotor.
- R_e es la resistencia eléctrica del motor.
- K_m es la constante de torque del motor.
- K_b es la constante de fuerza contraelectromotriz del motor.
- J_m es el momento rotacional total.
- R es el radio de las ruedas de tracción.
- m es la masa del vehículo.
- B_m es el coeficiente de fricción viscosa total.
- γ_1, γ_2 son constantes de distribución de la masa.
- V_1, V_2 son los voltajes de alimentación de los motores.

El modelo anterior se emplea para simular el AGV, de manera que al aplicarle cierto voltaje a los motores del mismo, se puede determinar con el modelo la velocidad angular que se producirá en cada rueda. Así pues, los motores reales son substituidos por su modelo y se asume que la velocidad angular calculada con las ecuaciones será la velocidad angular producida en los motores con el nivel de voltaje aplicado.

Con las ecuaciones del modelo obtenido y las ecuaciones de la trayectoria generada por las B-splines se establece una relación, dada por las ecuaciones cinemáticas del sistema, para realizar el control de seguimiento de la trayectoria para el vehículo. Las ecuaciones cinemáticas y el control de seguimiento se muestran en la sección siguiente.

4.4 SEGUIMIENTO DE LA TRAYECTORIA B-SPLINE.

El seguimiento de la trayectoria generada por las B-splines aprovecha la información implícita de estas curvas paramétricas descritas anteriormente y el modelo matemático del vehículo. El vínculo que se da entre las ecuaciones de la trayectoria de las B-splines y las ecuaciones del modelo matemático está dado por las ecuaciones cinemáticas.

En sistemas mecánicos una ligadura es una restricción al movimiento de un cuerpo [22]. “Una ligadura se puede expresar en términos de las condiciones que deben cumplir las coordenadas de un objeto o bien las restricciones de velocidad del cuerpo” [25]. Además, cuando una ligadura se puede expresar en términos geométricos el sistema se denomina holónimo. El término holónimo se refiere a que el número total de grados de libertad controlables y el número de grados de libertad totales del sistema es el mismo. Si se tiene una restricción de velocidad y esta se puede reescribir en función de coordenadas, entonces el sistema también es holónimo [22].

Las ruedas del modelo del vehículo que se utilizan describen un sistema no holónimo. Debido a que las ruedas se desplazan paralelamente sobre el plano entonces se necesitan cuatro coordenadas geométricas para describir su posición y su orientación. Con las coordenadas

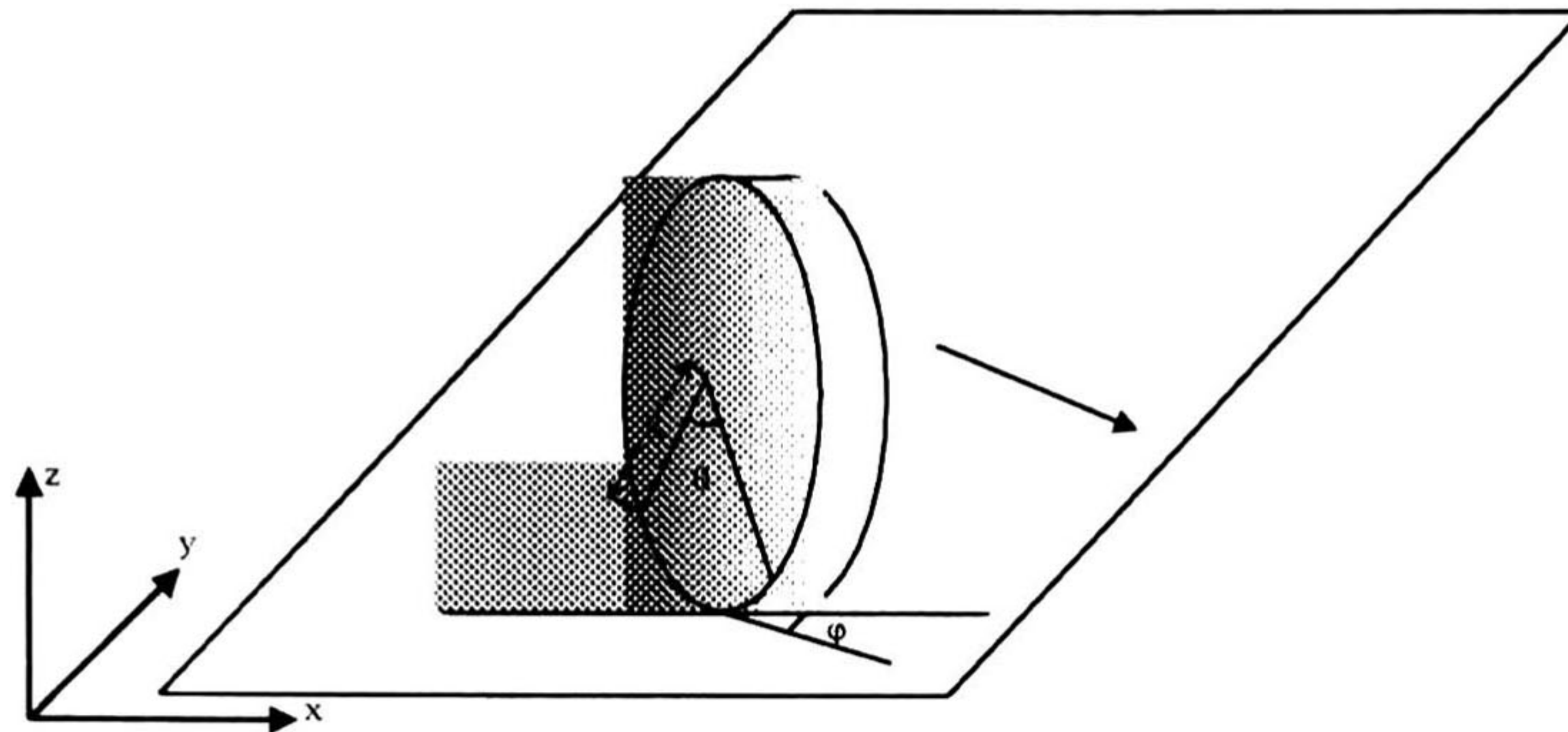


Figura 4.11: Posición y orientación de las ruedas del vehículo.

x-y se puede localizar la posición del punto de contacto de la rueda sobre el plano, el ángulo θ indica la posición angular de un punto de la rueda en relación con un punto de referencia de la rueda misma y el ángulo φ indica la orientación de la rueda con respecto al eje x. (Ver la figura 4.11).

Despreciando el deslizamiento de una rueda al rodar tenemos las dos ecuaciones de ligaduras siguientes [25]:

$$\begin{aligned} \dot{x} \operatorname{sen} \varphi + \dot{y} \cos \varphi &= 0 \\ \dot{x} \cos \varphi - \dot{y} \operatorname{sen} \varphi &= a \dot{\theta} \end{aligned}$$

La primera de las ecuaciones anteriores indica que la rueda no se puede mover a lo largo de su eje de giro. La segunda describe la velocidad de su centro. De estas dos ecuaciones restrictivas de velocidad no es posible obtener restricciones en términos de las coordenadas geométricas x , y , φ y θ , por lo que el sistema es no holónomo.

El número de grados de libertad de este sistema es de dos, ya que cada rueda es libre de deslizarse sobre el plano sin resbalar y de girar solo sobre su eje vertical. Cada rueda tendrá su propia velocidad lineal instantánea e independiente dada por:

$$\begin{aligned} v_d &= \omega_d R \\ v_i &= \omega_i R \end{aligned}$$

Además se tienen los elementos de arco alrededor del punto C, como se muestra en la figura 4.12.

De la figura anterior se puede determinar, (ver el desarrollo en [25]), que:

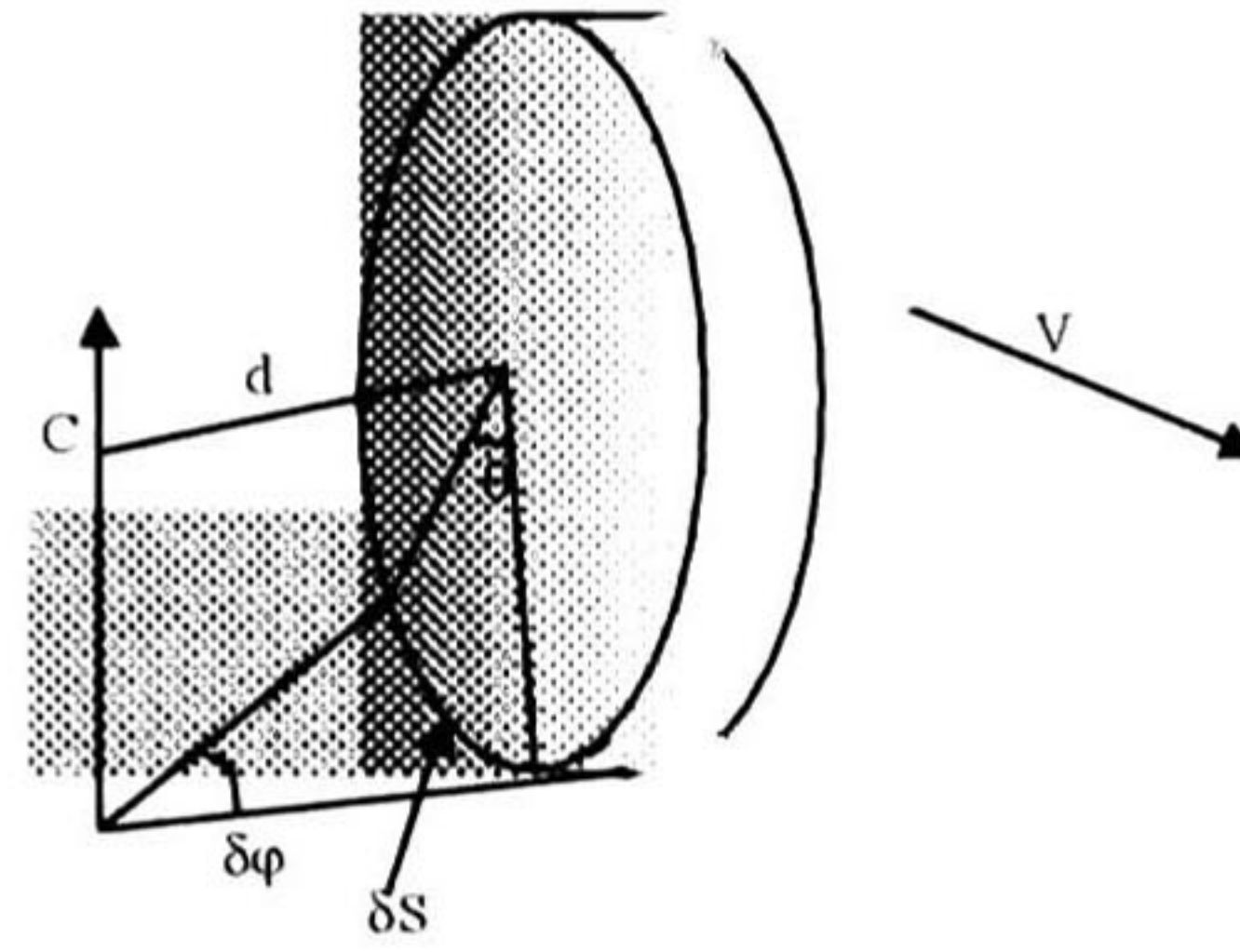


Figura 4.12: Elementos diferenciales de la cinemática de una rueda.

$$\dot{\varphi} = \frac{(\omega_d - \omega_i)R}{2d} \quad (4.18)$$

donde d es el diámetro de la rueda.

Considerando el promedio de las velocidades lineales de cada rueda como la velocidad resultante tenemos:

$$v = \frac{v_d + v_i}{2}$$

y en términos de las velocidades angulares tenemos:

$$v = \frac{(\omega_d + \omega_i)R}{2} \quad (4.19)$$

Los componentes de velocidad de las ruedas se pueden expresar como sigue (ver desarrollo en [25]):

$$\dot{x} = v \cos \varphi$$

$$\dot{y} = v \operatorname{sen} \varphi$$

Por lo tanto se tienen las ecuaciones siguientes:

$$\begin{aligned} \dot{x} &= \frac{(\omega_d + \omega_i)R}{2} \cos \varphi \\ \dot{y} &= \frac{(\omega_d + \omega_i)R}{2} \operatorname{sen} \varphi \end{aligned} \quad (4.20)$$

Una vez establecidas estas ecuaciones, junto con las ecuaciones de la trayectoria generada por las B-splines y las ecuaciones del modelo del vehículo se pueden determinar las ecuaciones para el control del seguimiento de la trayectoria generada.

La trayectoria generada es determinada por las B-splines. Los puntos de control de las B-splines que generan la trayectoria son los nodos de la ruta óptima determinada en el grafo de conectividad. La trayectoria se va generando durante el recorrido mientras se incrementa el parámetro t .

A partir de las ecuaciones 4.18 y 4.19 se obtienen las velocidades angulares de referencia, dadas por:

$$\begin{aligned}\omega_d^r &= \frac{v+\dot{\varphi}d}{R} \\ \omega_i^r &= \frac{v-\dot{\varphi}d}{R}\end{aligned}$$

De las B-splines que definen la trayectoria se obtiene el valor $\dot{\varphi}$ al derivar la ecuación 4.12 de las B-splines, obteniendo:

$$\dot{\varphi} = -\frac{\frac{\dot{y}\ddot{x}}{(\dot{x})^2} - \frac{\ddot{y}}{\dot{x}}}{1 + \frac{(\dot{y})^2}{(\dot{x})^2}}$$

Los términos de las derivadas para determinar $\dot{\varphi}$ en la ecuación anterior se obtienen de las ecuaciones 4.10 al aplicar un método de aproximación de derivadas de diferenciación numérica [16] en el programa simulador.

Con las velocidades angulares de referencia y las velocidades angulares del vehículo, determinadas con el modelo matemático de la ecuación 4.17, se implementa un control proporcional que determina el nivel de voltaje requerido en las terminales de cada uno de los motores para alcanzar las velocidades angulares de referencia, ω_d^r y ω_i^r . Es decir:

$$V = K_p(\vec{\omega}^r - \vec{\omega}).$$

Este voltaje es el aplicado a los motores del vehículo, que en el caso del simulador corresponde al voltaje V aplicado al modelo matemático de los motores en la ecuación 4.17.

Finalmente, la trayectoria recorrida por el vehículo se obtiene al integrar las ecuaciones 4.20, como sigue:

$$\begin{aligned}x &= \int \frac{(\omega_d + \omega_i)R}{2} \cos \varphi \\ y &= \int \frac{(\omega_d + \omega_i)R}{2} \operatorname{sen} \varphi\end{aligned}$$

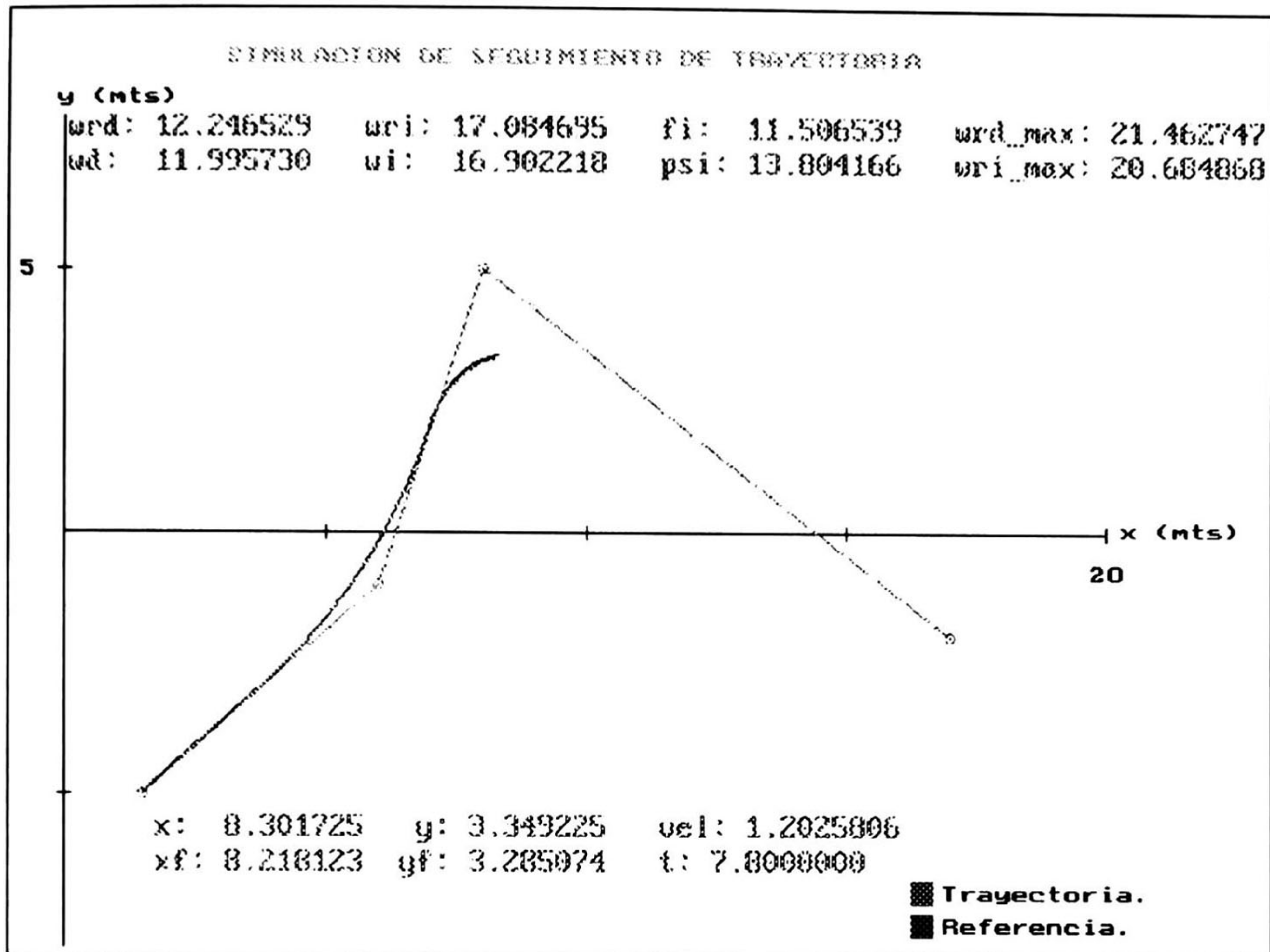


Figura 4.13: Seguimiento de una trayectoria en el programa simulador.

donde las velocidades angulares se obtienen del modelo del vehículo, considerando el nivel de voltaje aplicado en las terminales de los motores. En un vehículo real, estas velocidades angulares corresponderían a las velocidades angulares reales de los motores, que se pueden determinar con sensores. Las figuras 4.13 y 4.14 muestran un ejemplo de seguimiento de trayectoria en dos instantes diferentes del recorrido.

En el control del seguimiento de la trayectoria se emplea un control proporcional simple, por lo que la ganancia K_p puede determinar un mejor seguimiento de la misma al incrementarse. Sin embargo, al incrementarse excesivamente se pueden obtener niveles de voltaje muy altos, en comparación con los niveles disponibles para un vehículo real, que serían adecuados solo en simulación. Para un vehículo real la ganancia K_p debe ser ajustada de manera que no se sobrepase el límite de voltaje en los motores durante el seguimiento de la trayectoria. El algoritmo implementado en la generación y seguimiento de la trayectoria se resume en el Apéndice 1.

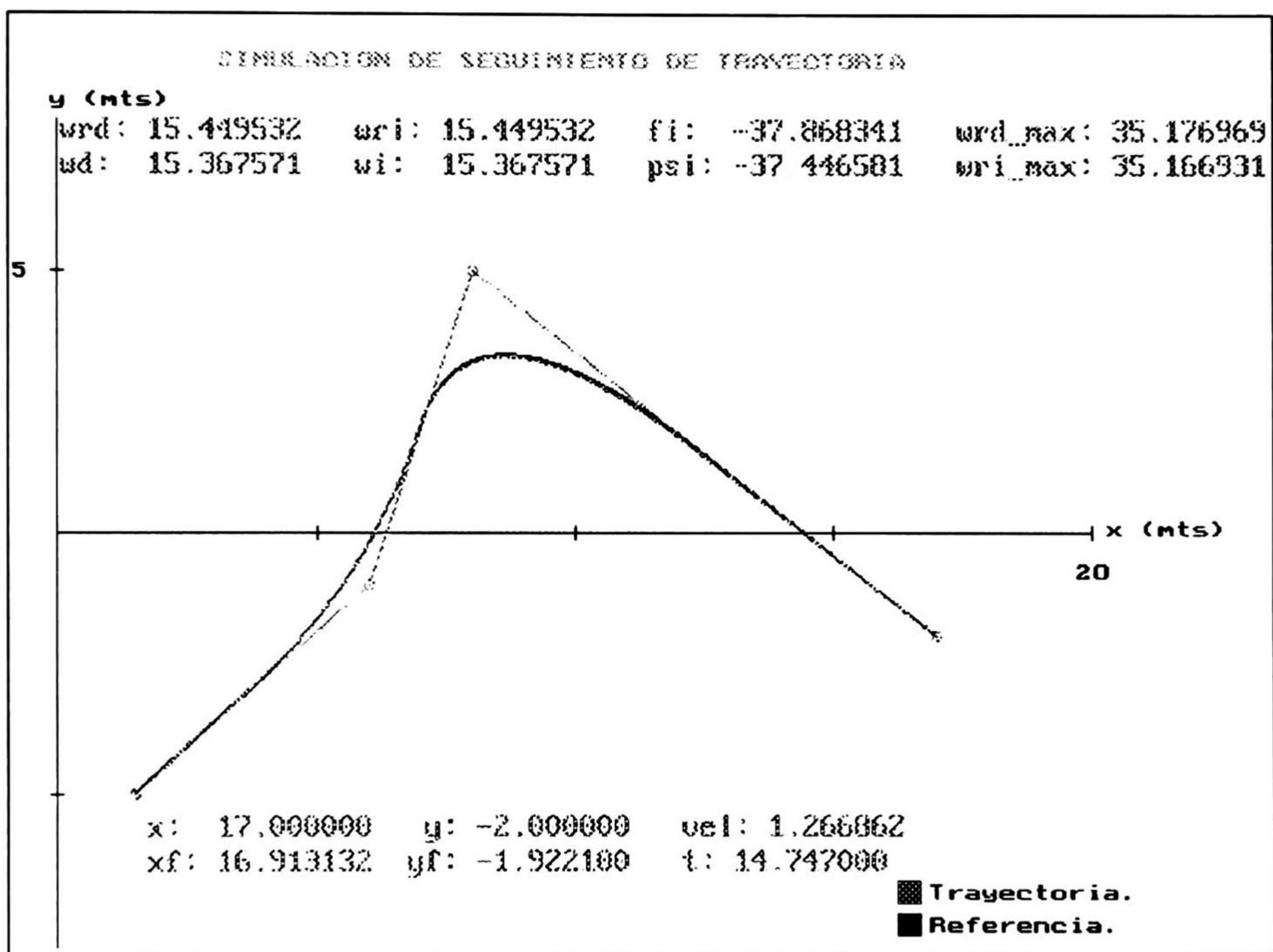


Figura 4.14: Seguimiento de una trayectoria en el programa simulador.

Capítulo 5

TAREAS REACTIVAS.

En el capítulo anterior se define la forma de generar la trayectoria del vehículo a partir de la ruta definida en el grafo de conectividad, que se presenta en el capítulo 3. En este capítulo se supone que el vehículo va generando la trayectoria durante su recorrido y al detectar un obstáculo que le impida seguir la trayectoria generada en un punto dado, se tendrá que modificar ésta para evadir el obstáculo. En este capítulo se verá el tema de las tareas reactivas, en las cuales se muestran características de detección de obstáculos de los sensores de ultrasonido simulados, y también la forma en que se modifica la trayectoria durante el recorrido para evadir los obstáculos que se interponen en el camino.

Las tareas reactivas en un AGV comprenden múltiples tareas de acciones que realiza el vehículo ante situaciones no consideradas en la preplaneación ante las cuales el AGV reacciona de acuerdo con las indicaciones especificadas por sus tareas reactivas. Entre estas tareas se encuentran alarmas de baterías bajas, modificación de trayectorias ante obstáculos, alarmas de camino bloqueado, alarmas y tareas de pérdida de la carga, etc. En esta sección mostraremos la manera en que puede ser modificada la trayectoria del vehículo al ser detectado un obstáculo que lo obstruya en su recorrido.

A continuación se presentan las características de los sensores de ultrasonido que detectan los obstáculos, y se indica más adelante la forma en que éstos son detectados en el simulador. Finalmente, se mostrará la manera en que se modifica la trayectoria para permitir al vehículo evadir los obstáculos detectados.

5.1 SENSORES.

Las tareas reactivas en un AGV dependen en gran medida de los sensores que detectan la situación de "peligro" ante la cual el vehículo debe reaccionar. El tipo de sensores considerados en el simulador del AGV para detectar los obstáculos en su trayectoria son sensores de ultrasonido.

Todos los sensores sean activos o pasivos, desarrollan su función detectando, y en la mayoría de los casos cuantificando, el cambio en alguna propiedad específica de energía. Los sensores que miden la distancia hacia un objetivo de interés sin contacto físico directo se pueden referir como sensores de rango sin contacto. Hay al menos siete técnicas diferentes empleadas en varios dispositivos de medición de distancia [8]:

- Triangulación.
- Tiempo de vuelo (pulsos).
- Mediciones de corrimiento de fase (CW).
- Modulación de frecuencia.
- Interferometría.
- Barrido de foco.
- Intensidad de la señal de retorno.

Los sensores de ultrasonido emiten una señal que viaja a través del aire y que es reflejada por objetos localizados dentro del rango del sensor, al ser reflejada la señal en un objeto, cierta parte de energía es regresada hacia el sensor y detectada por un transductor receptor. Estos sensores emplean la técnica de tiempo de vuelo para determinar la distancia a la cual se encuentran los objetos que reflejan la señal acústica emitida. Los sensores de ultrasonido emplean señales acústicas de frecuencias típicamente entre 20 KHz y 200 KHz; mientras que los rangos audibles son de 20 Hz a 20 KHz. Para todo sistema acústico se debe considerar el medio de propagación que en algunos casos puede tener una influencia significativa en las mediciones tomadas. El sonido es una perturbación mecánica vibratoria que viaja a través de un medio elástico como una onda longitudinal. Para gases y líquidos la velocidad de propagación de onda está dada por [18]:

$$s = \sqrt{\frac{k_m}{\rho}}$$

donde :

s es la velocidad de propagación.

k_m es el módulo de elasticidad de masa.

ρ es la densidad del medio.

La longitud de onda de la señal acústica es directamente proporcional a la velocidad de propagación y está dada por:

$$\lambda = \frac{s}{f}$$

donde :

λ es la longitud de onda.

s es la velocidad del sonido.

f es la frecuencia de operación.

La velocidad del sonido en el aire, considerando una temperatura de 20 °C y al nivel del mar, es de 340 m/seg. La distancia a la que se encuentra un objeto se puede determinar conociendo el tiempo de vuelo de la señal. Considerando que la velocidad de la señal del sensor viaja a la velocidad del sonido, y que el tiempo de vuelo corresponde al tiempo total de ida y de regreso de la señal, la distancia a un objeto detectado se puede calcular por la fórmula siguiente:

$$d_{ob} = \left(170 \frac{m}{seg}\right)(t_v) \quad (5.1)$$

donde:

t_v es el tiempo de vuelo de la señal.

La acción de los sensores de ultrasonido fue implementada en el simulador para detectar obstáculos. Se realizaron pruebas y experimentos para poder registrar el comportamiento de los sensores de ultrasonido reales. En la tabla 5.2 se muestran los parámetros de los sensores empleados, también llamados *trascptores ultrasónicos* ya que contienen transductores que transmiten y reciben la señal.

PARAMETROS	VALOR	UNIDADES
Voltaje de Alimentación	5 a 10	Volts (CD)
Consumo de Corriente	25	mA
Ángulo de Operación	±12	Grados
Frecuencia y Ancho de Banda	40±1	KHz
Dimensiones	3.6x1.2x1.1	Pulgadas
Rango (Alcance)	0.4 (0.1) a 66 (20)	Pies (Metros)

Las señales de los sensores de ultrasonido empleados en las pruebas se presentan en la figura 5.1. La *señal de control* es la emitida por el sensor y en la *salida digital* del sensor se recibe el eco al reflejarse en un objeto. El tiempo de vuelo es $t_v = T_1 - T_0$. La distancia de un objeto dentro del rango del sensor se determina empleando la ecuación 5.1.

Los sensores de ultrasonido son adecuados para la detección de obstáculos ya que pueden detectar obstáculos de madera, metálicos, plásticos, etc. Sin embargo, existen algunos factores que pueden afectar las mediciones. Estos factores se presentan en la sección siguiente.

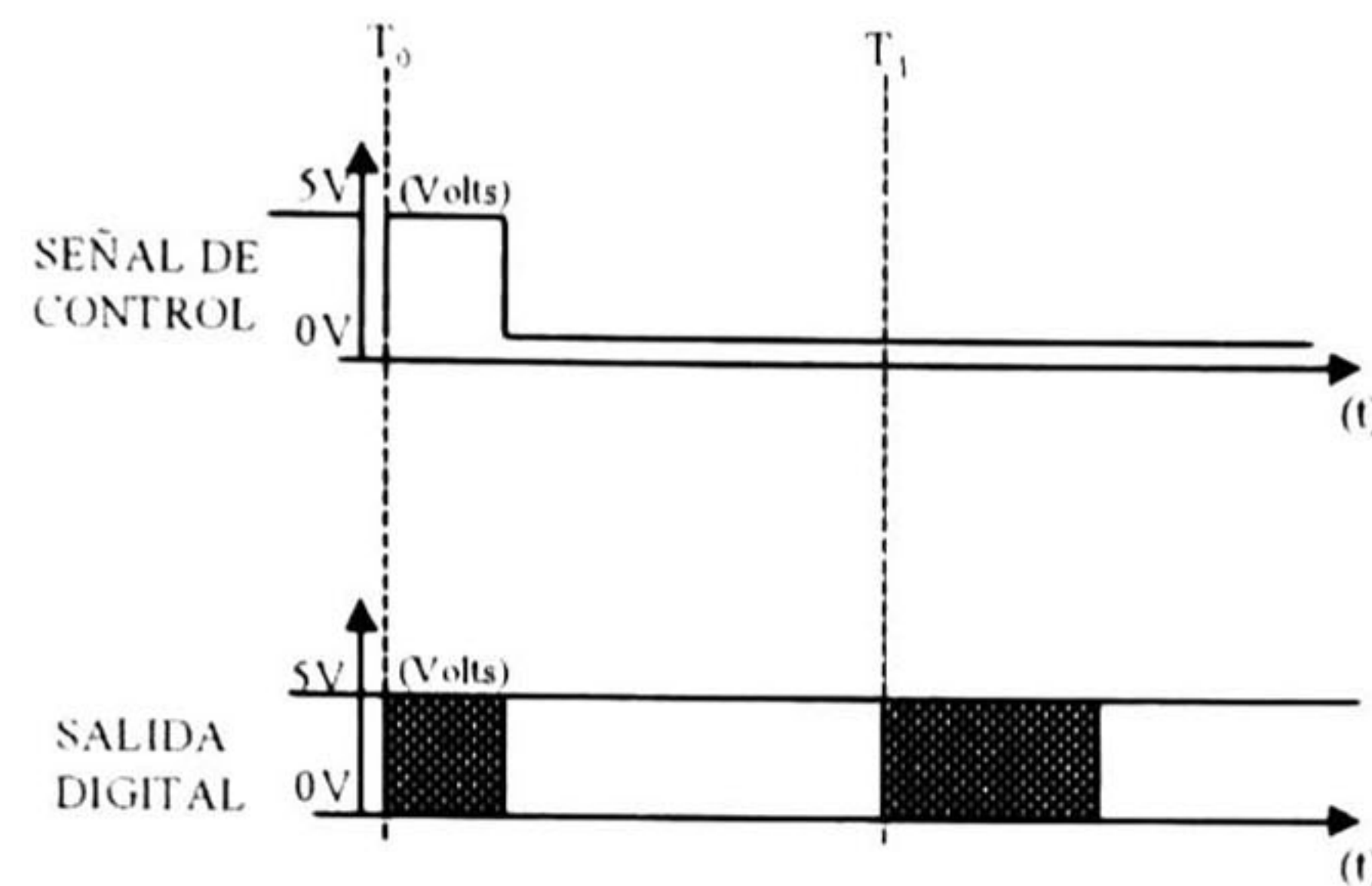


Figura 5.1: Señales de los sensores de ultrasonido.

5.2 DETECCIÓN DE OBSTÁCULOS EN LA TRAYECTORIA.

La detección de obstáculos en la trayectoria del AGV es realizada por los sensores de ultrasonido que emiten señales acústicas hacia el medio ambiente y detectan la señal cuando es reflejada por algún objeto que está dentro del rango del sensor. Si los sensores no reciben ningún eco de la señal emitida entonces se puede suponer que no hay obstáculos próximos que obstruyen al vehículo.

La señal incidente en un objeto es en su totalidad reflejada o absorbida, ya sea de naturaleza acústica, óptica o de radio frecuencia. La dirección de la superficie del objeto determina la cantidad de señal reflejada hacia el transductor receptor. La mayoría de los objetos se pueden considerar como lisos en comparación con las relativamente largas longitudes de onda de la energía ultrasónica. En este caso, el ángulo de reflexión es igual al ángulo de incidencia. Así que, cuando un ángulo de incidencia del abanico del sonar¹ es menor que cierto ángulo crítico, la energía reflejada no regresa al transductor receptor (ver la figura 5.2). Este ángulo crítico está en función de la frecuencia de operación y de las características geométricas de la superficie reflejante.

Algunos factores que afectan a las mediciones son provocados por el ambiente de trabajo, tales como la atenuación de la señal en la atmósfera, la turbulencia del aire, la temperatura y el ruido de alguna otra fuente externa en el mismo espacio de trabajo. Otros factores determinantes en la detección de obstáculos corresponden a las características propias de los sensores, tales como el ángulo de su abanico y su alcance. Y otros más, a la configuración y distribución de los sensores en el vehículo.

¹Un sonar es un transductor que emite una señal acústica de frecuencia fija cuyo eco puede ser detectado por algún transductor receptor.

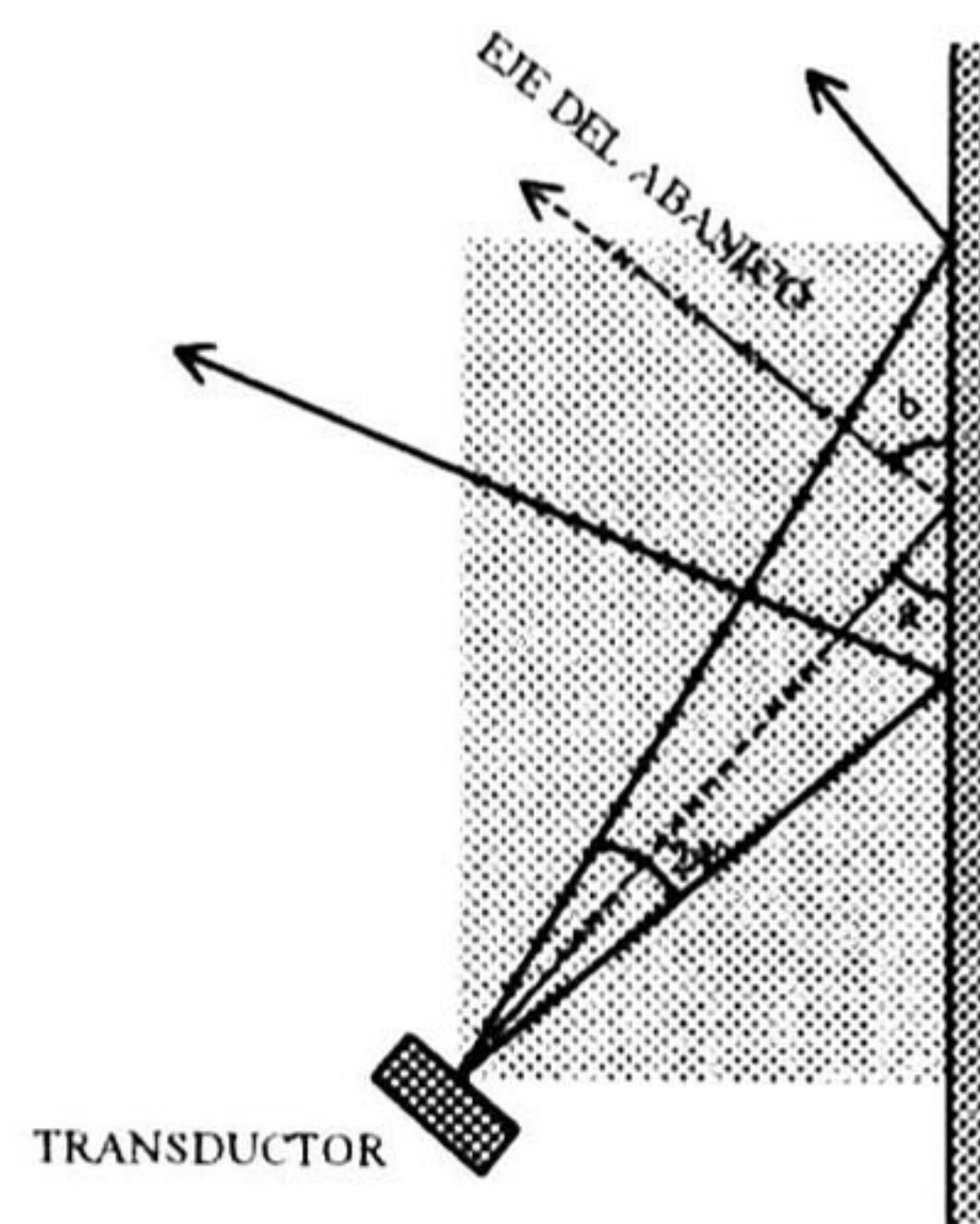


Figura 5.2: Ángulo de incidencia menor al ángulo crítico.

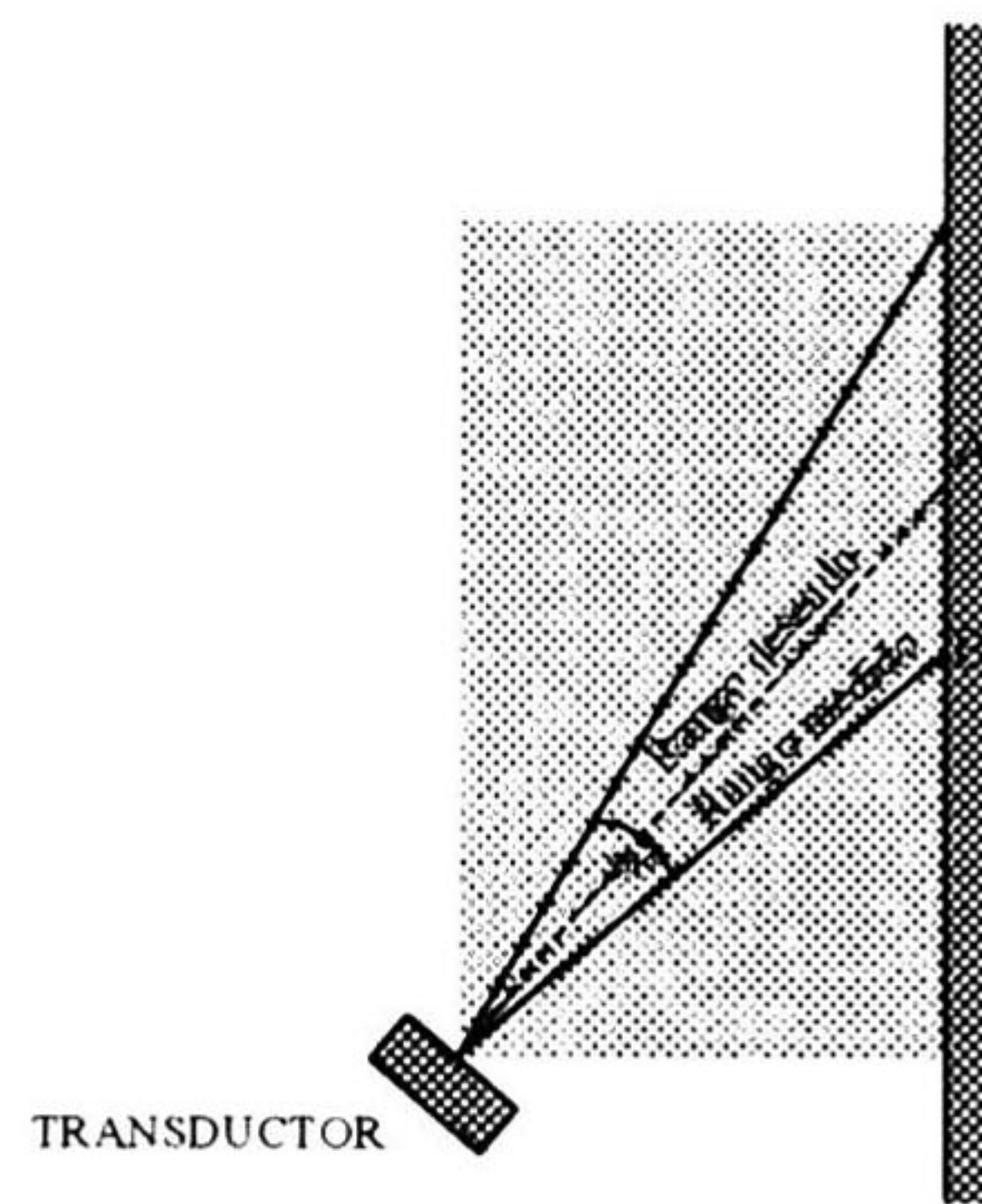


Figura 5.3: Discrepancia entre las distancias medida y deseada.

Esta claro que los mejores resultados en la detección de obstáculos se obtienen cuando la línea central del abanico del sensor se mantiene normal a la superficie del objeto detectado. Mientras el ángulo de incidencia varía de la perpendicularidad, se puede notar que la distancia medida en realidad no siempre corresponde con aquella asociada con la línea central del abanico (ver la figura 5.3).

Existen configuraciones para mejorar la resolución angular que consisten en compartir el abanico que generan dos sensores de manera que se traslapan las señales de ambos. La figura 5.4 muestra este principio para dos sensores, de manera que si ambos sensores A y B detectan el obstáculo entonces éste o parte de éste está en la región de traslape que corresponde al área sombreada. Si el obstáculo es detectado por A pero no por B, entonces éste se encuentra en la región superior del abanico del sensor A. Aumentando el número de sensores con abanicos traslapados se disminuye el tamaño de las regiones respectivas y se incrementa la resolución angular.

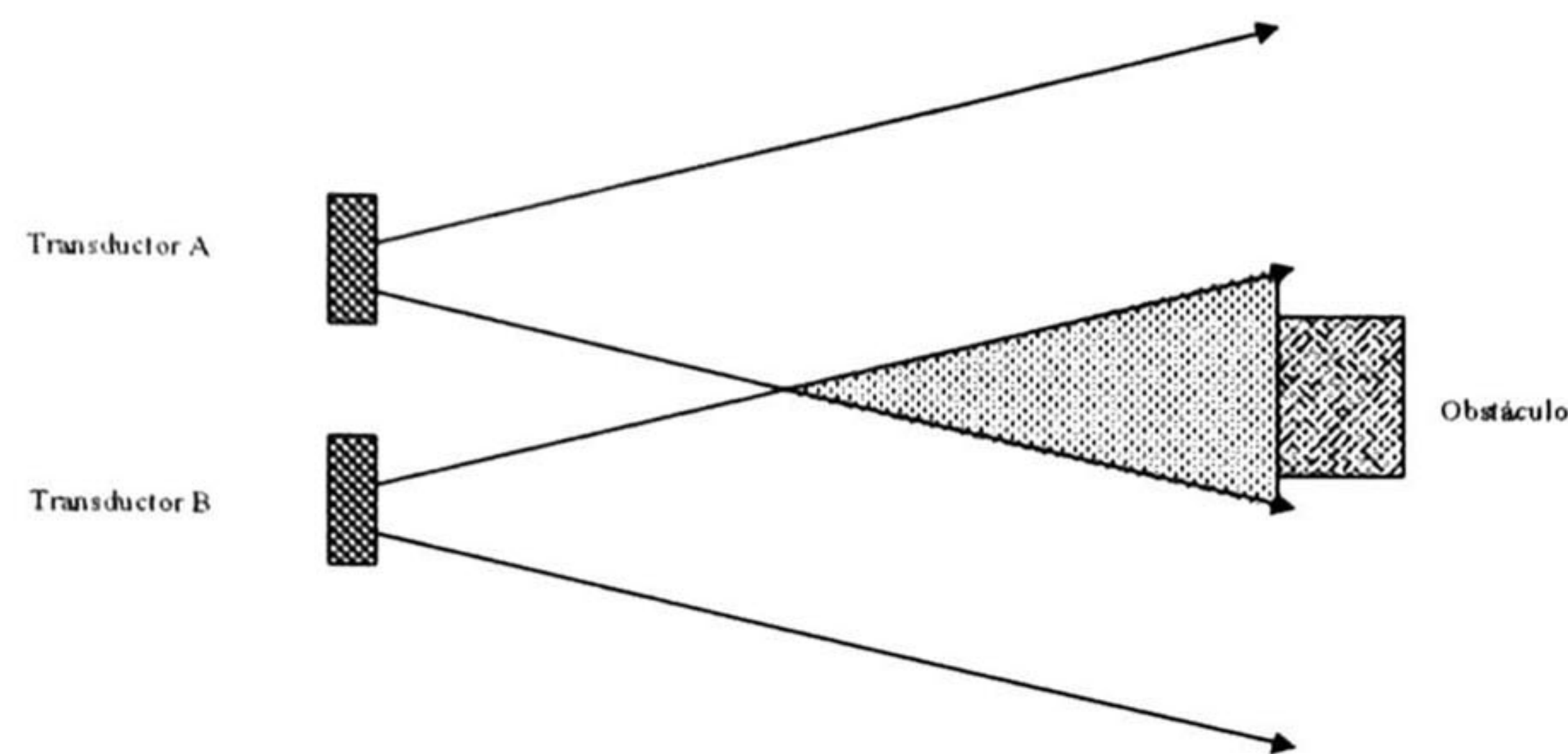


Figura 5.4: Técnica de traslape de abanicos con dos sensores.

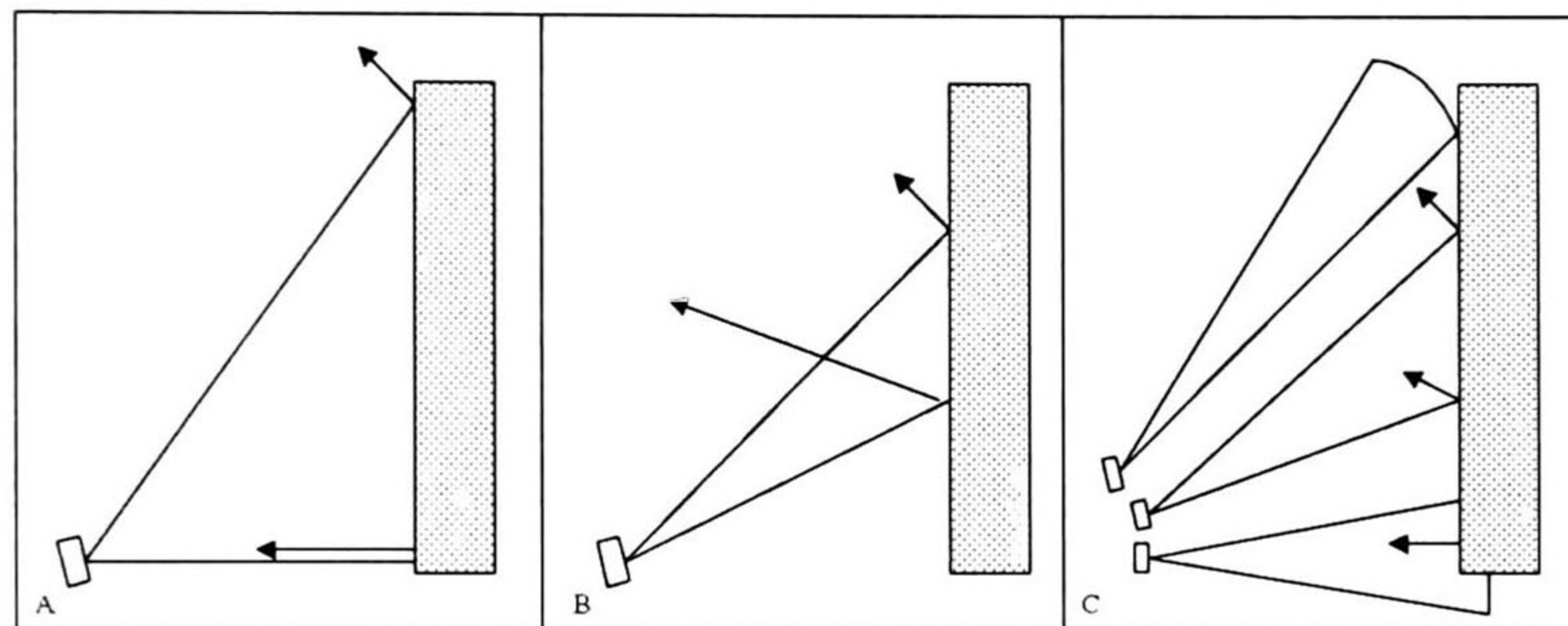


Figura 5.5: Arreglos de sensores.

El uso de sensores con un abanico de ángulo mayor puede mejorar la detección de obstáculos debido a que es más probable que alguna porción del abanico encuentre una superficie normal como se observa en la figura 5.5. Es decir, que mientras más abierto sea el abanico hay mayor oportunidad de encontrar un obstáculo, pero al mismo tiempo se pierde la precisión en la ubicación del obstáculo. El transductor en A tiene un abanico más abierto y detecta el obstáculo que no es detectado por el transductor en B. Una alternativa es utilizar un arreglo cilíndrico de transductores de abanico angosto que tengan la misma cobertura pero que al detectarse un obstáculo sea más precisa la ubicación del mismo, como se muestra en C. Al tener diferentes sensores en un mismo arreglo se puede presentar ruido entre las señales cruzadas de unos con otros por lo que se debe tener cuidado en la distribución y orientación de los sensores en el vehículo.

La señal de los sensores del simulador representan una aproximación al comportamiento de sensores reales que fueron probados en experimentos de laboratorio. Durante la experimentación se realizó un circuito electrónico de control para el disparo de los sensores y la obtención de información de la señal recibida por los transductores receptores. Este circuito se utilizó como la interface entre los sensores y un programa de control de sensores, pro-

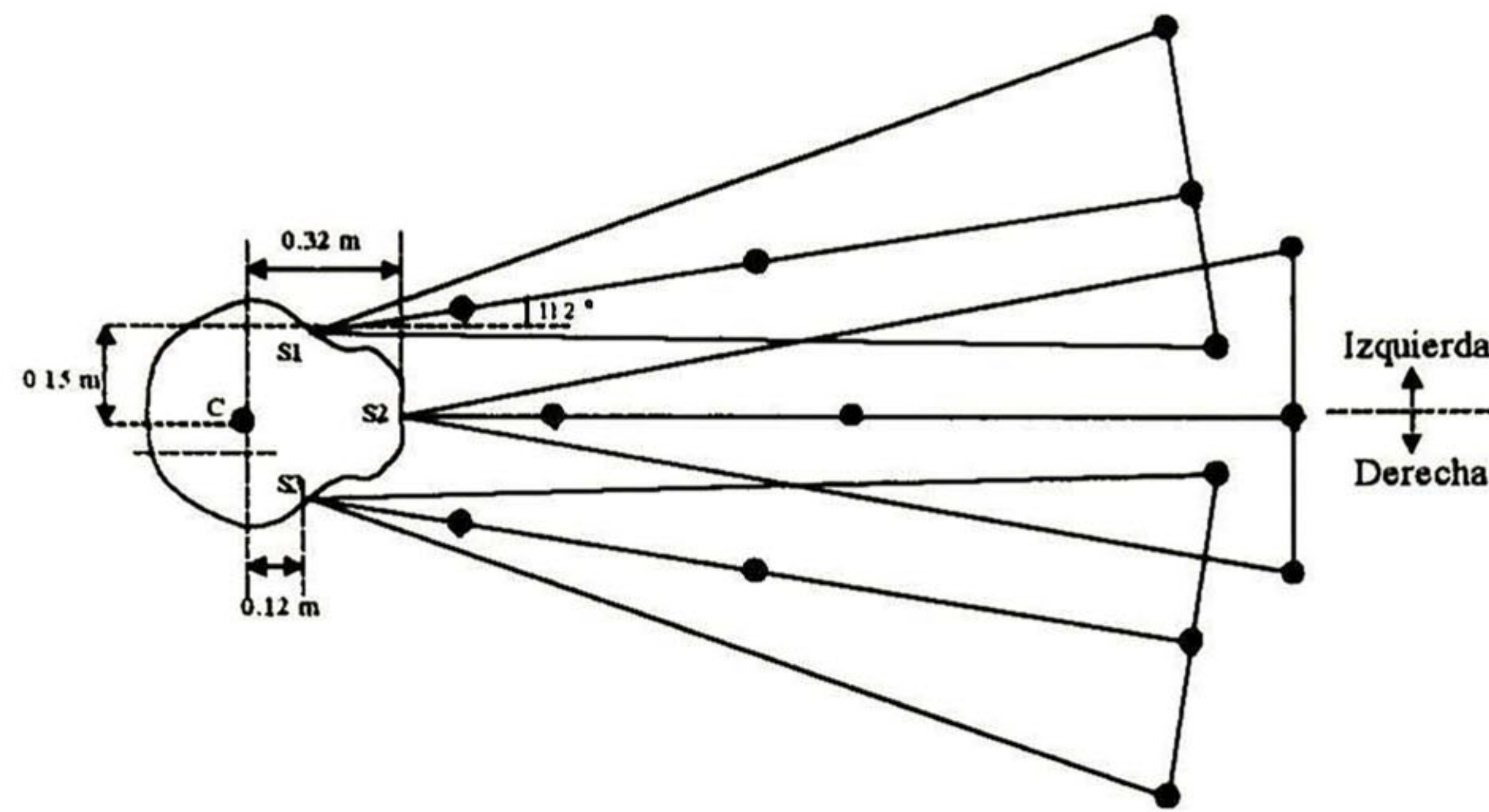


Figura 5.6: Distribución de sensores en el frente del vehículo.

gramado en C++. El programa maneja a los sensores, es decir, determina el disparo de la señal del sensor y realiza un monitoreo de la señal del transductor receptor para identificar un objeto detectado por el sensor. Además, en él se realiza el cálculo de la distancia entre el objeto detectado y el sensor, analizando el tiempo de vuelo de la señal. El tiempo de vuelo es el tiempo de retardo entre la señal de control y la señal digital del transductor receptor (ver la figura 5.1). El objetivo de estas pruebas fue conocer y analizar el funcionamiento de los sensores de ultrasonido reales, para representar la detección de obstáculos en el programa simulador. En las pruebas realizadas se utilizó solamente un sensor a la vez, por lo que no se trató con problemas de integración sensorial. En el programa simulador no se considera el problema de señales cruzadas entre los sensores.

En el vehículo simulado se consideran tres sensores localizados en la parte frontal del vehículo. La distribución de los sensores se muestra en la figura 5.6.

El punto C corresponde al centro del eje axial de los motores y las coordenadas para el vehículo en el control de seguimiento son dadas para el punto de referencia C.

Las consideraciones tomadas en los sensores para simular la detección de obstáculos en el simulador son las siguientes (ver la figura 5.7):

1. El abanico del sensor tiene una distancia máxima de detección dt_{\max} de 1.5 mts.
2. La amplitud total del abanico de detección ad_t se determina por el ángulo de operación de $\pm 12^\circ$, y es de 0.64 mts.
3. En el abanico de detección de obstáculos de cada sensor se identifican 5 puntos llamados *puntos de detección* pd_j con $j = 1, \dots, 5$.
4. Tres de los puntos de detección son localizados a lo largo del eje central del abanico a 0.25, 0.75 y 1.5 mts del sensor.
5. Los dos puntos de detección restantes se localizan en los extremos del abanico a 0.32 mts del eje central.

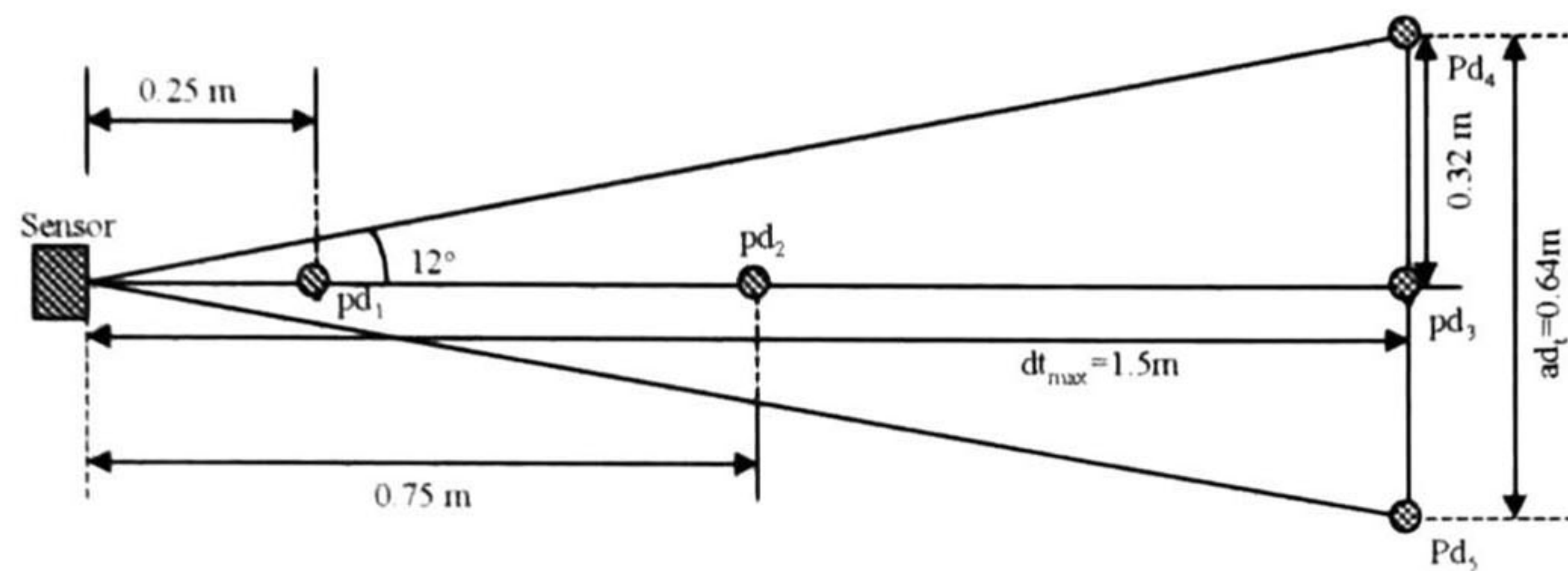


Figura 5.7: Localización de los *puntos de detección* en el abanico del sensor.

La detección de obstáculos en el simulador se lleva a cabo cuando el área del abanico de un sensor se traslapa con el área ocupada por un obstáculo. Los puntos de detección ayudan a identificar cuando el abanico cruza la región de un obstáculo, de manera que un obstáculo será detectado por un sensor en el simulador cuando:

1. Los puntos de detección pd_4 o pd_5 junto con el punto pd_3 estén dentro del área de un objeto. En este caso se denomina *detección a distancia media*.
2. Cuando el punto de detección pd_2 este dentro del área de un objeto. A este caso se denomina *detección a distancia cercana*.
3. Cuando el punto de detección pd_1 este dentro del área de un objeto. En este caso se denomina *detección a distancia crítica*.

La distancia a la que se encuentra un obstáculo detectado por un sensor se determina calculando la distancia entre el borde del obstáculo (más próximo al vehículo) y el sensor, sobre el eje del abanico del sensor.

El tipo de obstáculo simulado es un objeto cuadrado de $1m^2$ que se puede mover con el teclado concurrentemente con el seguimiento de la trayectoria del AGV, para el estudio del comportamiento de evasión de obstáculos. La trayectoria del vehículo será modificada al detectarse un obstáculo por los sensores, evitando la colisión. La trayectoria se modifica dependiendo de la ubicación del obstáculo, que puede ser aproximada dependiendo del sensor que lo detectó. La modificación de la trayectoria para la evasión de obstáculos se presenta en la sección siguiente.

5.3 MODIFICACIÓN DE LA TRAYECTORIA.

Un número considerable de diferentes estrategias de control de navegación han sido desarrolladas y adoptadas para la evasión de obstáculos en los robots móviles. En estas estrategias se pueden diferenciar las estrategias de *navegación global* y las de *navegación local*. En las técnicas de navegación global se tiene una etapa de planeación que establece una ruta óptima

hacia alguna posición meta en un sistema de coordenadas. En la navegación local el robot navega en medio de obstrucciones inesperadas.

La llamada navegación global es la parte del simulador discutida en los capítulos 3 y 4, donde logramos obtener una trayectoria para el AGV. En las tareas reactivas, la evasión de obstáculos corresponde a la navegación local en la que el vehículo es capaz de evadir los obstáculos que obstruyen su trayectoria. Las estrategias de navegación local se pueden subdividir en tres categorías básicas: *control reactivo*, *modelado representativo del mundo* y alguna combinación de ambos. A continuación se describen algunos métodos de cada una de estas categorías.

CONTROL REACTIVO.

Las técnicas de control reactivo se refieren a estrategias de comportamiento basado que acoplan directamente la información sensorial en tiempo real con acciones en los motores, sin el uso de representaciones simbólicas que intervengan en un intento de modelar en un sentido absoluto o parcial el ambiente de operación del robot. El control reactivo presenta las siguientes características generales [9]:

- Generalmente se manifiesta por una descomposición en comportamientos básicos.
- Se evitan representaciones globales.
- Se prefiere un desacoplamiento de sensores sobre una fusión de sensores.
- Es muy conveniente para ambientes dinámicos cambiantes.

Entre los métodos de control reactivo podemos encontrar las estrategias siguientes [9]:

A) Rutina "Vaga" En este tipo de navegación se describe un comportamiento primitivo que consiste en viajar en línea recta hasta que se encuentra un obstáculo, con lo que se altera el curso para evitar la colisión, y después retomando un movimiento en línea recta [7, 6, 2, 1].

B) Circumnavegación. El término *circumnavegación* describe un comportamiento de evasión de obstáculos en el cual el robot se desplaza lateralmente para evadir una obstrucción, mientras que se mantiene en la dirección hacia la meta [9].

C) Campos Potenciales. El método involucra la presencia de una fuerza artificial que actúa sobre el robot, derivada de una sumatoria de vectores de una fuerza atractiva representada por la meta y un número de fuerzas repulsivas asociadas con los obstáculos individuales conocidos. Este método se describe en el capítulo 2.

D) Esquema Motor. Este método emplea una estrategia de evasión de obstáculos que es muy similar al de los campos potenciales. Los esquemas son básicamente comportamientos primitivos que representan comportamientos motores parametrizados. Una colección de tales esquemas proporciona una familia de acciones para el control del robot [3, 4].

MODELADO REPRESENTATIVO DEL MUNDO.

Las técnicas de *modelado representativo del mundo* involucran un acoplamiento apropiado de sensores con algún tipo de capacidad de modelado del “mundo” exterior. Los esfuerzos de estas técnicas se dirigen hacia las tareas siguientes [9]:

- Sensores que colectan datos de los objetos en los alrededores del robot.
- Construcción apropiada de un modelo del mundo exterior.
- Planeación de la trayectoria deseada.
- Evasión de obstáculos durante el recorrido.

La principal ventaja de estos métodos sobre las estrategias puramente reactivas es que garantizan encontrar una trayectoria libre hacia la meta, siempre que esta trayectoria exista.

MÉTODOS COMBINADOS.

Existen métodos que emplean algunas de las técnicas del control reactivo junto con técnicas de los métodos de modelado representativo del mundo. Estos métodos son de alguna manera implementaciones innovadoras que emplean combinaciones de ambas metodologías para obtener operaciones más robustas en ambientes dinámicos.

La estrategia de evasión de obstáculos propuesta en el simulador es una técnica combinada, que emplea un modelo de representación del mundo que es determinado por un grafo; haciendo uso al mismo tiempo de esquemas simples de evasión de obstáculos.

La manera de implementar el modelo representativo del mundo para evadir los obstáculos detectados por el vehículo simulado es modificando la ruta del grafo de conectividad durante el recorrido. Dado que la trayectoria se va generando, a partir de los nodos que forman la ruta mientras el vehículo se mueve hacia la configuración final, entonces si agregamos puntos de control de B-splines para definir la trayectoria al detectar un obstáculo modificaremos también la trayectoria original de manera que se evite la colisión. Así pues, la evasión de obstáculos dependerá de la ubicación de los nuevos puntos de control para las B-splines que corresponden a *nodos auxiliares* en la ruta definida en el grafo de conectividad. Para ilustrar una modificación de la trayectoria por medio de los nodos auxiliares supongamos

que el vehículo tiene una trayectoria original recta como se ve en la figura 5.8 al detectar un obstáculo. En el momento de la detección del obstáculo se asignan los nodos auxiliares N_{aux} que provocan un cambio en la trayectoria. Esto se logra por tres propiedades fundamentales de las B-splines (dadas en la sección 4.2) : 1) Los puntos nudo se encuentran en el punto medio de dos puntos de control. 2) El segmento que une dos puntos de control es tangente a la trayectoria en el punto nudo en medio de ellos. 3) La velocidad en un punto nudo está dada por $\dot{\vec{P}}_i (t = 0) = \vec{Q}_i - \vec{Q}_{i-1}$ o $\dot{\vec{P}}_i (t = 1) = \vec{Q}_{i+1} - \vec{Q}_i$. Estas tres propiedades permiten asegurar que la trayectoria cruce por un punto deseado (un punto nudo), con una dirección deseada y a una velocidad también especificada. La ubicación de los nodos N_{aux} debe cumplir con las tres reglas siguientes:

1. Se deben ubicar dos primeros nodos N_{aux} alineados en la dirección de la tangente en el punto C, de referencia del vehículo; de manera que el punto C coincida con el nuevo punto nudo formado en el punto medio de los dos nodos auxiliares, como lo muestra la figura 5.8. De esta manera aseguramos una continuidad en la trayectoria del vehículo.

2. La distancia a la que se ubican los dos primeros nodos auxiliares del punto de referencia C, debe ser:

$$d_{aux} = \frac{|\dot{\vec{P}}_i|}{2} \quad (5.3)$$

Donde $\dot{\vec{P}}_i$ corresponde a la curva de la trayectoria original en el momento de detectarse el obstáculo. Con esta condición aseguramos la continuidad en la velocidad y se satisface $\dot{\vec{P}}_i (t = 0) = \vec{Q}_i - \vec{Q}_{i-1}$ o $\dot{\vec{P}}_i (t = 1) = \vec{Q}_{i+1} - \vec{Q}_i$ para el nuevo punto nudo. En la figura 5.8 se muestra la ubicación de los dos primeros nodos auxiliares.

3. Se deben asignar otros nodos auxiliares que permitan evadir el obstáculo detectado. Para ello se localiza uno o más puntos Pl del espacio libre por los cuales al pasar el vehículo se evita la colisión y se asignan dos nodos N_{aux} por cada punto Pl escogido, de manera que aseguren que el vehículo cruzará por Pl , con la dirección deseada y a la velocidad requerida. En la figura 5.8 se muestra la ubicación de los nodos auxiliares para un punto libre Pl .

Los esquemas simples que determinan los movimientos de modificación de la trayectoria están dados por nueve movimientos de reacción del vehículo ante distintas situaciones de detección de obstáculos. En la tabla 5.4 se muestran los casos en que se especifica cada uno de los movimientos propuestos para el simulador. Para cada uno de los movimientos se tiene un arreglo de los nodos auxiliares que definen la modificación de la trayectoria.

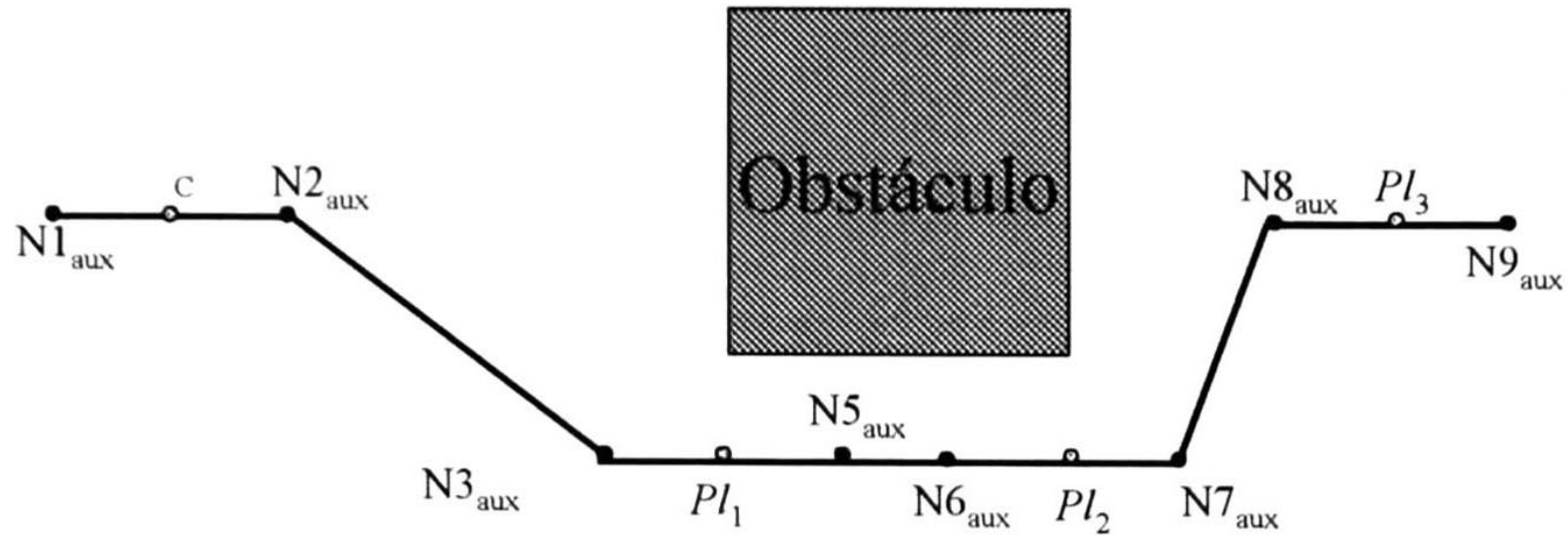


Figura 5.8: Ubicación de los nodos auxiliares para los nodos Pl asignados.

DISTANCIA DE DETECCIÓN	No. DE SENSORES	SENSOR ID.	MOVIMIENTO DE REACCIÓN
Media	1 Sensor	S1	Ninguno
		S2	Ninguno
		S3	Ninguno
	2 Sensores	S1 y S2	MOV1-Derecha
		S1 y S3	Ninguno
		S2 y S3	MOV1-Izquierda
3 Sensores	S1, S2 y S3	MOV2-X	
Cercana	1 Sensor	S1	MOV3-Derecha
		S2	MOV4-X
		S3	MOV3-Izquierda
	2 Sensores	S1 y S2	MOV4-Derecha
		S1 y S3	Paro
		S2 y S3	MOV4-Izquierda
	3 Sensores	S1, S2 y S3	Paro
Crítica	Cualquiera	Cualquiera	Paro

(5.4)

En la tabla anterior se especifican los nueve tipos de movimientos de reacción que debe ejecutar el vehículo para evadir el obstáculo detectado por los sensores. Los términos Izquierda y Derecha toman como referencia el eje central del abanico del sensor S2, como lo muestra la figura 5.6. Los movimientos marcados con X especifican un movimiento hacia la Izquierda o hacia la Derecha dependiendo de la convexidad de la curva B-spline o de la trayectoria original. Si la curva de la trayectoria hace mover al vehículo en la dirección Derecha, entonces X representará un movimiento hacia la Izquierda, y viceversa. A continuación se describen los movimientos y la ubicación de los nodos auxiliares que permiten realizar un movimiento determinado:

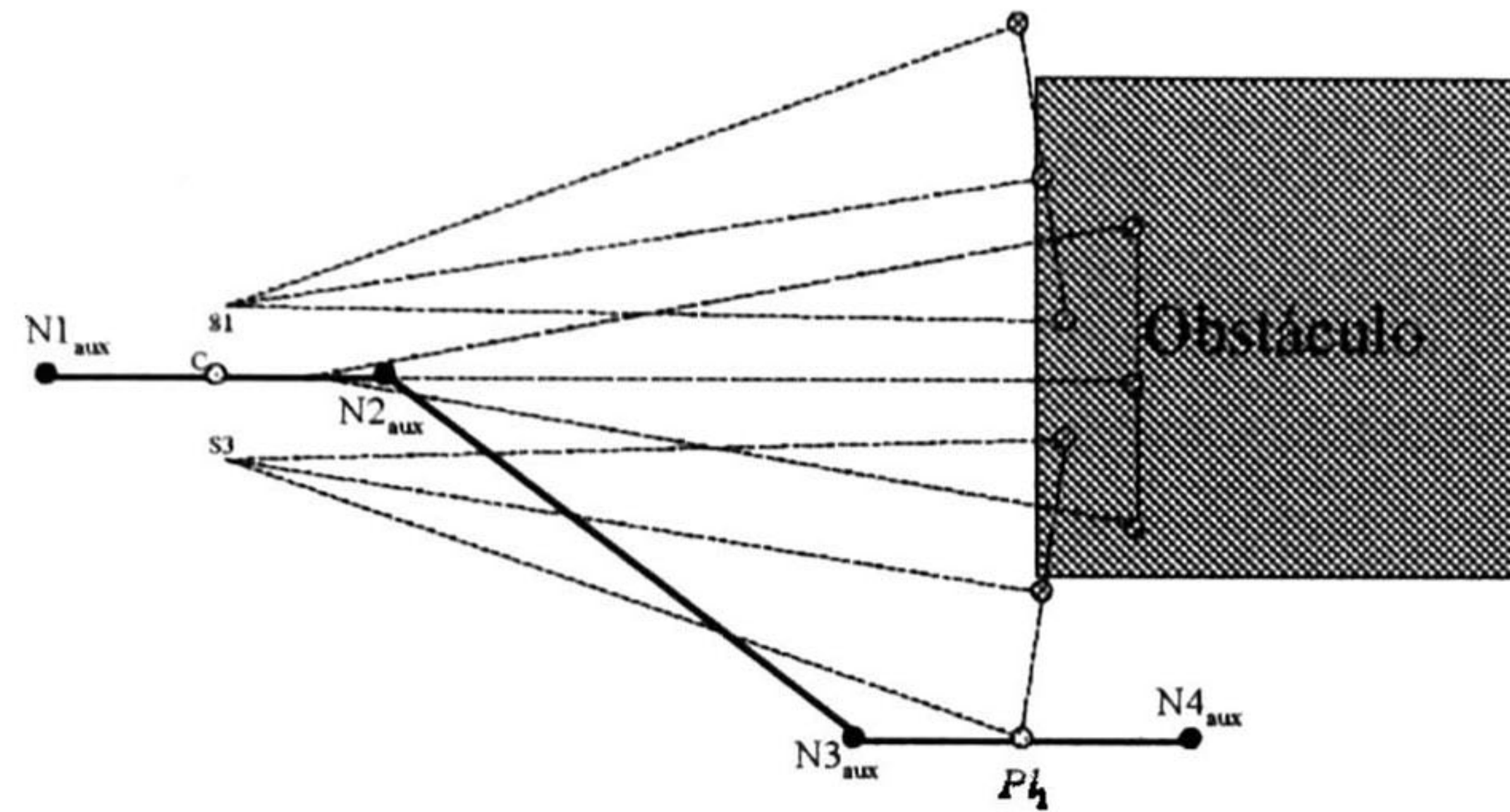


Figura 5.9: Movimiento de reacción MOV1-Derecha para evadir el obstáculo.

Movimiento MOV1-Derecha/Izquierda.

En el movimiento hacia la derecha el vehículo evade el obstáculo, detectado por los sensores S1 y S2, al pasar a través del punto pd_5 del sensor S3. Para lograrlo los nodos auxiliares se localizan como lo muestra la figura 5.9, que representa un movimiento hacia la derecha. La distancia de separación de los nodos auxiliares se determina por la ecuación 5.3.

En el movimiento a la izquierda los nodos se ubican en la posición “espejo” con respecto al eje central del abanico S2.

Movimiento MOV2-Derecha/Izquierda.

En el movimiento a la derecha se ubican inicialmente tres nodos auxiliares que hacen girar el vehículo, como se muestra en la figura 5.10. Una vez que el vehículo gira lo suficiente para que el obstáculo deje de ser detectado por el sensor S3, se ubican nuevos nodos auxiliares para cruzar el punto pd_5 del sensor S3, como en el movimiento MOV1-Derecha, pero respetando la dirección inicial del movimiento cuando el obstáculo fue detectado por los tres sensores.

En el movimiento a la izquierda los nodos se ubican en la posición “espejo” con respecto al eje central del abanico S2.

Movimiento MOV3-Derecha/Izquierda.

En el movimiento hacia la derecha el vehículo evade el obstáculo al pasar a través del punto pd_3 del sensor S3. Los nodos auxiliares son ubicados como lo muestra la figura 5.11.

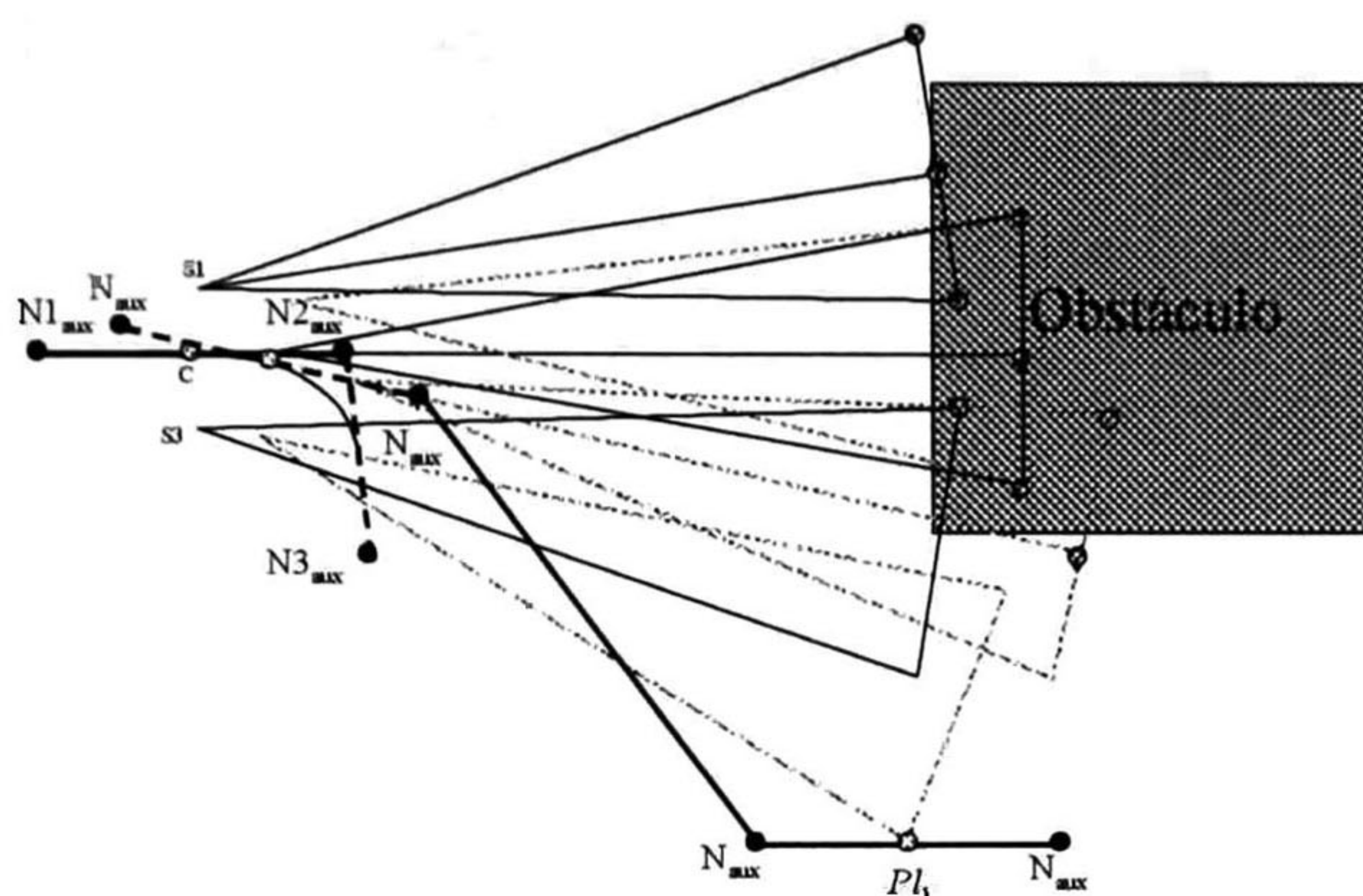


Figura 5.10: Movimiento de reacción MOV2-Derecha.

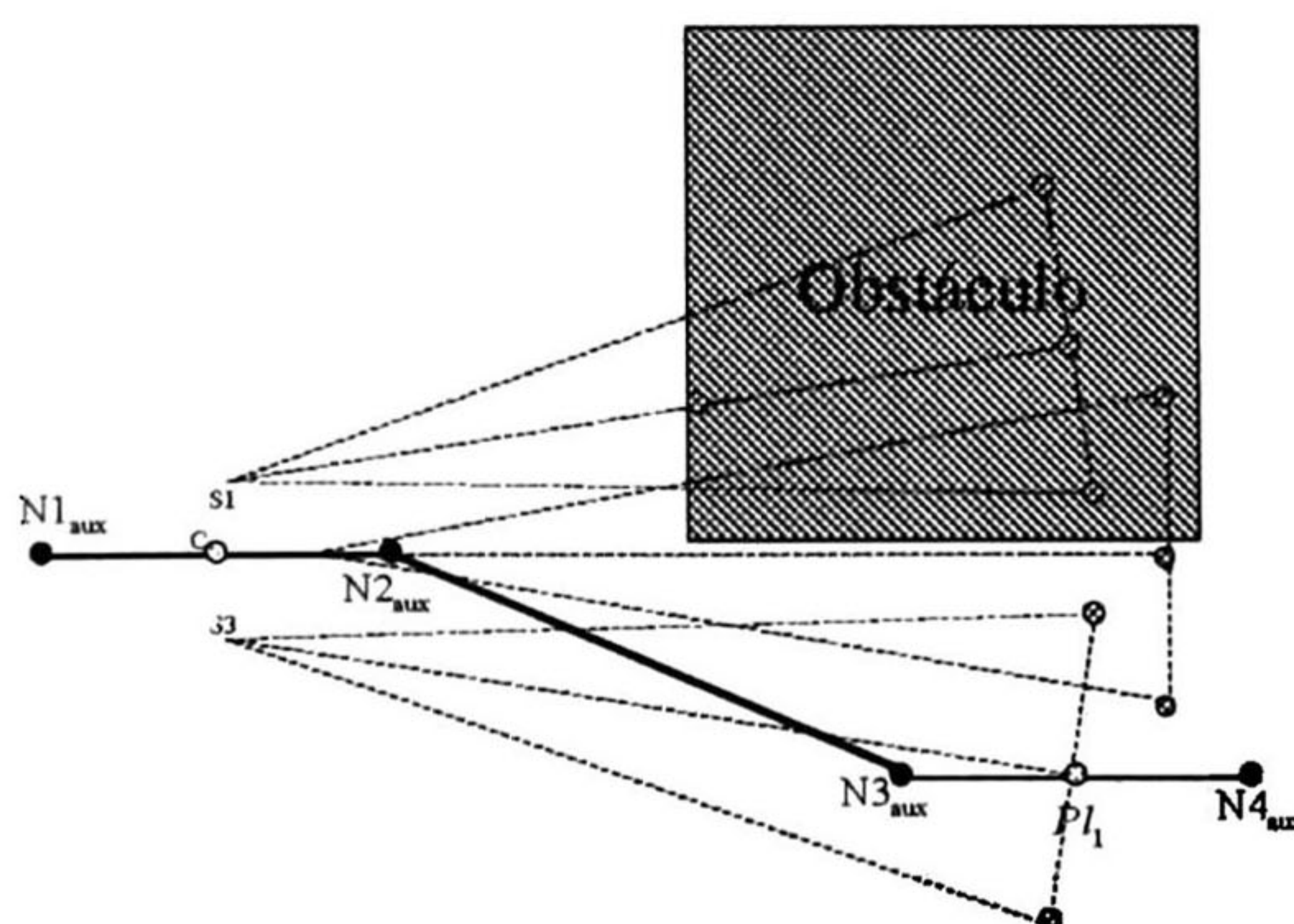


Figura 5.11: Movimiento de reacción MOV3-Derecha.

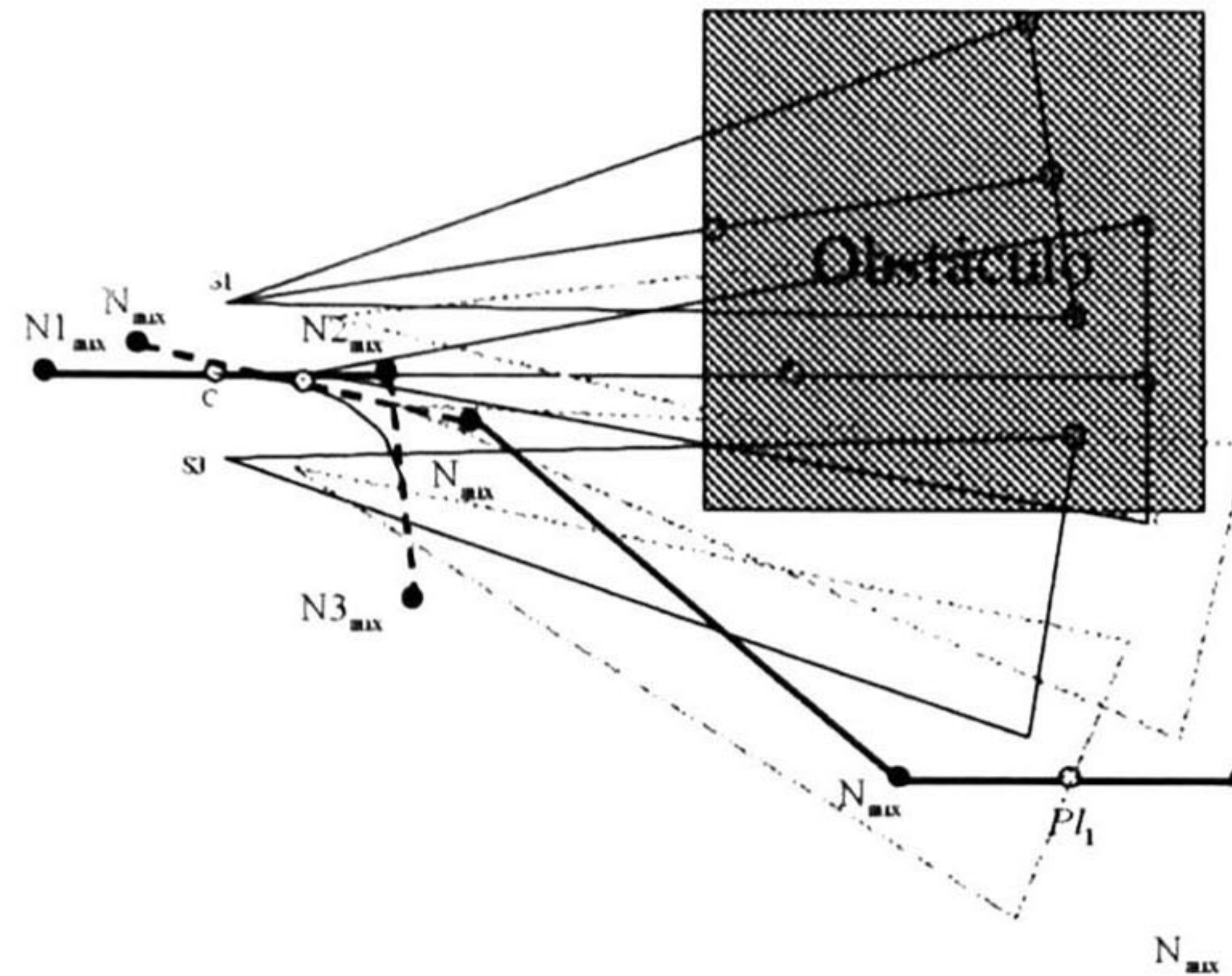


Figura 5.12: Movimiento de reacción MOV4-Derecha.

En el movimiento a la izquierda los nodos se ubican en la posición “espejo” con respecto al eje central del abanico S2.

Movimiento MOV4-Derecha/Izquierda.

En el movimiento a la derecha se ubican inicialmente tres nodos auxiliares que hacen girar el vehículo, como se muestra en la figura 5.12. Una vez que el vehículo gira lo suficiente para que el obstáculo deje de ser detectado por el sensor S2, se ubican nuevos nodos auxiliares para cruzar el punto pd_3 del sensor S3, como en el movimiento MOV3-Derecha, pero respetando la dirección inicial del movimiento cuando el obstáculo fue detectado por los dos sensores.

En el movimiento a la izquierda los nodos se ubican en la posición “espejo” con respecto al eje central del abanico S2.

Movimiento de Paro.

Esta reacción provoca que el vehículo se detenga completamente pues no tiene oportunidad de evadir el obstáculo detectado. Se puede activar en este momento una alarma auditiva para que el obstáculo sea removido o bien se puede reiniciar la búsqueda de una nueva ruta a partir de ese punto hacia la configuración final.

Si se decide buscar una nueva ruta se debe tomar en cuenta que el vehículo debe orientarse hacia la dirección de la nueva trayectoria.

Recuperación de la Trayectoria Original.

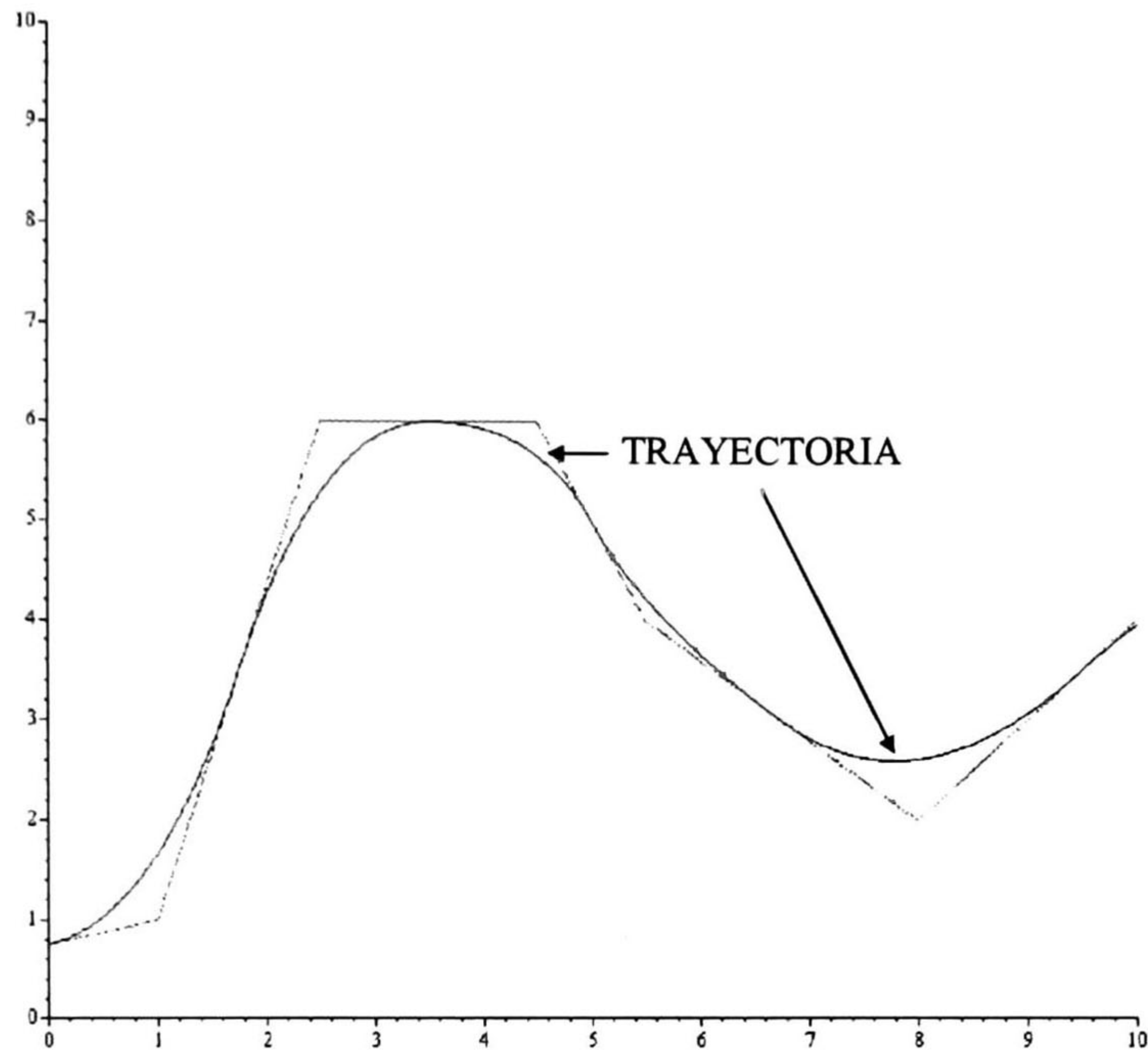


Figura 5.13: Seguimiento de trayectoria sin obstáculo.

La recuperación de la trayectoria original se logra por la propiedad de *Variabilidad* de las B-splines (sección 4.2), ya que si respetamos los siguientes nodos de control en la ruta del vehículo, la trayectoria se modifica solo localmente para evadir el obstáculo sin alterar el resto de la trayectoria original.

En la evasión de obstáculos con los esquemas de movimiento de reacción se considera cuatro nodos auxiliares para realizar la evasión como lo muestran las figuras correspondientes a cada uno de ellos. Si la modificación de la trayectoria ocurre en una curva $P_i(t)$, definida por los puntos de control \vec{Q}_{i-1} , \vec{Q}_i y \vec{Q}_{i+1} , entonces los puntos de control \vec{Q}_{i-1} y \vec{Q}_i son sustituidos por los nodos auxiliares. El punto de control \vec{Q}_{i+1} se mantiene para establecer el enlace con la trayectoria original en la recuperación de la trayectoria. Al llegar al último nodo auxiliar de los movimientos de reacción se asignarán dos nodos auxiliares más que garanticen que su punto medio Pl coincida con el punto nudo entre \vec{Q}_{i+1} y \vec{Q}_{i+2} . La figura 5.13 muestra un seguimiento de trayectoria en el cual no existen obstáculos. La figura 5.14 muestra la modificación de la trayectoria producida al detectarse un obstáculo.

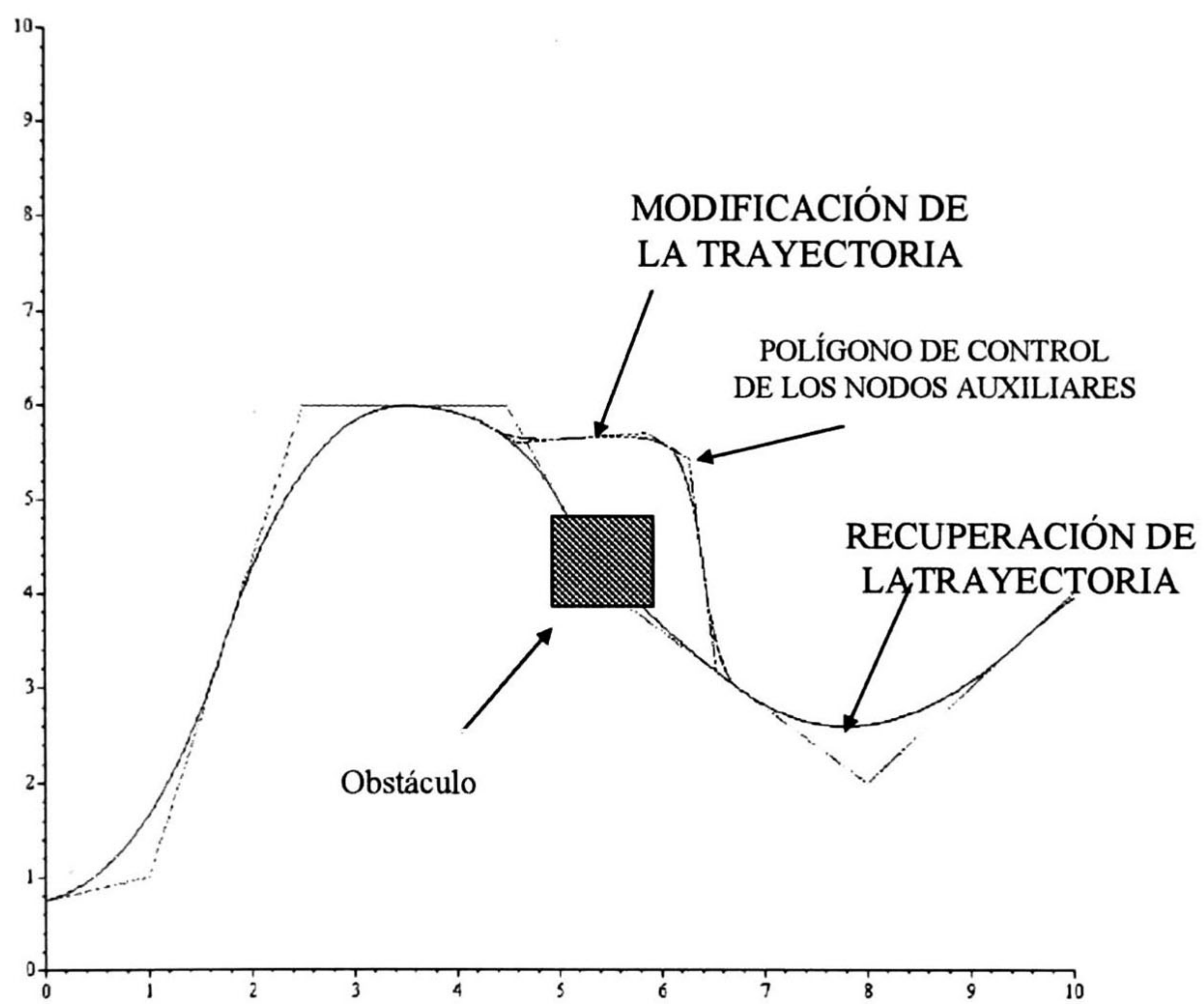


Figura 5.14: Simulación de modificación de la trayectoria para evadir el obstáculo y recuperación de la trayectoria original.

Capítulo 6

CONCLUSIONES Y TRABAJO FUTURO.

En el presente trabajo se estudiaron diferentes arquitecturas de AGV y se seleccionó una que resulta adecuada para los fines de investigación. Seguido se analizaron los diferentes módulos que comprenden un AGV y se implementó un simulador para probar los algoritmos que permiten la navegación de un AGV de forma autónoma. A continuación se listan, del vehículo, las tareas estudiadas en esta tesis y después se mencionan las conclusiones obtenidas durante el trabajo.

- La representación formal del espacio de trabajo del AGV por medio de grafos.
- Un algoritmo de generación del grafo de conectividad que representa el mapa del espacio libre del AGV.
- La implementación del algoritmo A* para determinar la ruta óptima.
- Un algoritmo de generación de trayectorias basado en curvas paramétricas B-splines para “suavizar” la ruta del AGV.
- Un algoritmo de seguimiento de trayectoria basado en un modelo matemático del AGV y la información implícita de las curvas B-splines que definen la trayectoria deseada.
- La simulación de detección de obstáculos por la acción de sensores de ultrasonido determinada experimentalmente.
- El desarrollo de un mecanismo capaz de modificar la trayectoria en base a las propiedades de las B-splines que la definen.
- Tareas reactivas de movimientos de evasión de obstáculos y recuperación de la trayectoria original.

La representación del conocimiento del espacio de trabajo por medio de grafos complementada con el método de búsqueda A^* , es una forma adecuada para resolver la etapa de preplaneación del movimiento en los AGV. Sin embargo, el reto principal en esta parte es generar un grafo que contenga la mayor información del espacio libre del vehículo posible, sin generar grafos extensos que compliquen la búsqueda de la ruta óptima. Para hacer frente a este problema, se propone un algoritmo de descomposición de celdas (polinomial en el número de obstáculos) que genera un grafo de conectividad satisfactorio para los fines del programa simulador.

La generación y seguimiento de la trayectoria se basa en las curvas B-splines, tomando los nodos de la ruta definida en el grafo de conectividad como sus puntos de control. Una limitante de las B-splines es que se complica el control de velocidad en la trayectoria del vehículo. El control empleado en el seguimiento de la trayectoria es un control proporcional. Los resultados obtenidos con el control implementado son considerados satisfactorios para nuestros fines, obteniendo errores de menos de 0.1 m en un recorrido total de 25 m. Sin embargo, se pueden emplear otro tipo de controles que permitan obtener un seguimiento más preciso de la trayectoria de referencia.

En la modificación de la trayectoria para evadir obstáculos en las tareas reactivas, se aprovechan las características de las B-splines. Sin embargo, resulta complicado establecer el control de evasión de obstáculos. De ahí que, consideramos que un algoritmo de evasión de obstáculos empleando las técnicas de campos potenciales sería más conveniente para este fin.

Finalmente, el programa simulador nos permite probar los algoritmos empleados y permite que se modifiquen en su código los parámetros del modelo del vehículo; siendo posible además, el intercambio de los algoritmos utilizados por nuevos algoritmos. Esto permite que en el mismo programa se puedan probar otros algoritmos de ruteo, generación de trayectorias, etc.

Los logros obtenidos durante el desarrollo de esta tesis y las pruebas realizadas han sido satisfactorios en base a los objetivos trazados al inicio del proyecto, considerando además las múltiples disciplinas que envuelven a las tareas del AGV. Sin embargo, al mismo tiempo surgen nuevos retos por concluir y que siguen abiertos para un trabajo futuro en este tema, como son:

- La implementación en el programa simulador del algoritmo de generación del grafo de conectividad propuesto.
- Dado que el seguimiento de trayectoria implementado requiere que el vehículo se encuentre en la posición y dirección de inicio de la trayectoria, es necesario un mecanismo que garantice la localización y orientación iniciales del vehículo antes de iniciar su recorrido.

- Solucionar las restricciones dinámicas en el radio de curvatura para vehículos con remolque.
- Introducir el control de velocidad a lo largo de la trayectoria para el AGV.
- Realizar pruebas de los programas de control implementados sobre un prototipo real.
- Realizar el control del vehículo con un programa en tiempo real.
- Probar las tareas reactivas con señales de sensores reales que involucren tiempos de muestreo y se observen problemas en la integración sensorial.

Los problemas abiertos mencionados anteriormente involucran múltiples disciplinas, los investigadores que trabajen en este tema pueden utilizar el programa simulador para probar sus algoritmos de control (ya sea de alto o bajo nivel) y acelerar su investigación.

APENDICE 1. DIAGRAMAS DE FLUJO DEL PROGRAMA SIMULADOR.

En este apéndice se muestran los diagramas de flujo del programa simulador. El código fuente del simulador fue realizado en lenguaje C++.

La figura 6.1 muestra el diagrama de flujo del algoritmo de generación del grafo de conectividad. Este es el primer algoritmo necesario en el control de navegación del AGV, con este algoritmo se genera el grafo que representa la información del mapa del espacio libre del vehículo. El algoritmo inicia con la información del mapa del área de trabajo del vehículo, donde se especifica la distribución de los salones y se localizan los obstáculos fijos, así como la posición de los puntos de carga y descarga para el vehículo. Este algoritmo no ha sido implementado en el programa simulador. El algoritmo de generación del grafo de conectividad se describe en el capítulo 3.

La figura 6.2 muestra el algoritmo A^* que define la ruta óptima en el grafo de conectividad. El programa simulador inicia con este algoritmo, así que necesita que se introduzca manualmente la ubicación de los nodos del grafo de conectividad y la relación de adyacencia entre ellos. El grafo de conectividad se puede generar con el método descrito en el capítulo 3 y los datos de éste se introducen en el programa como lo indica el apéndice 2.

Una vez que se establece la ruta por el algoritmo A^* , con la información de la posición de los nodos que forman la ruta, da inicio el algoritmo de generación y seguimiento de la trayectoria, que se muestra en la figura 6.3. En éste algoritmo, los nodos de la ruta son los puntos de control $\vec{Q}_0, \vec{Q}_1, \dots, \vec{Q}_n$, donde \vec{Q}_0 corresponde a la posición del nodo de inicio y \vec{Q}_n a la posición del nodo meta. Los puntos de control de las B-splines definen la trayectoria generada. En este algoritmo se integran las tareas reactivas que funcionan en paralelo con el algoritmo de generación y seguimiento de la trayectoria, y éstas originan una modificación en la trayectoria original al detectarse un obstáculo. El algoritmo 6.3 termina cuando el vehículo ha alcanzado la meta o es abortado por el usuario como se indica en apéndice 2.

Los algoritmos descritos forman los bloques que componen al programa simulador desarrollado. El simulador puede ser modificado en su programa fuente de manera que se integren

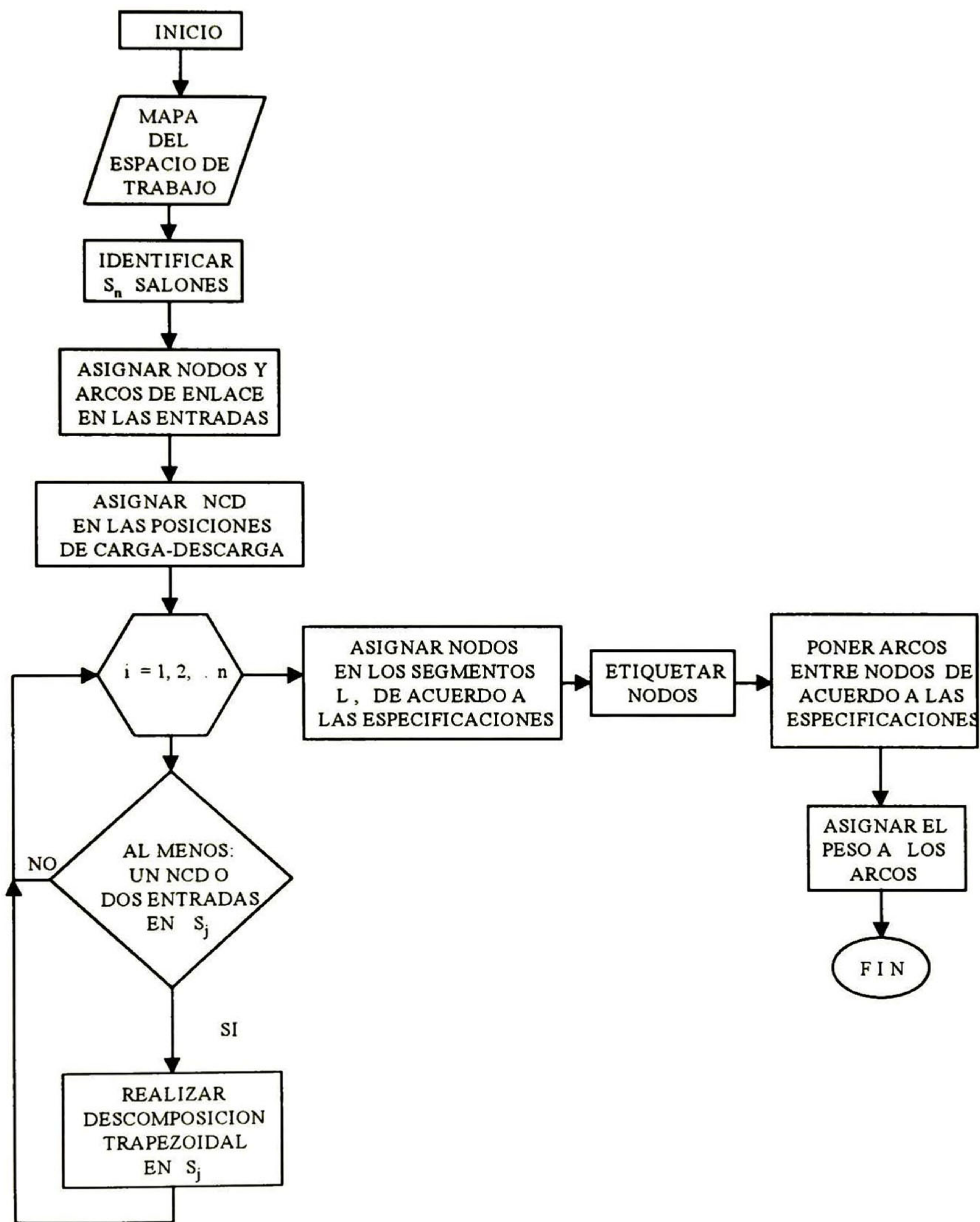


Figura 6.1: Algoritmo generador del grafo de conectividad.

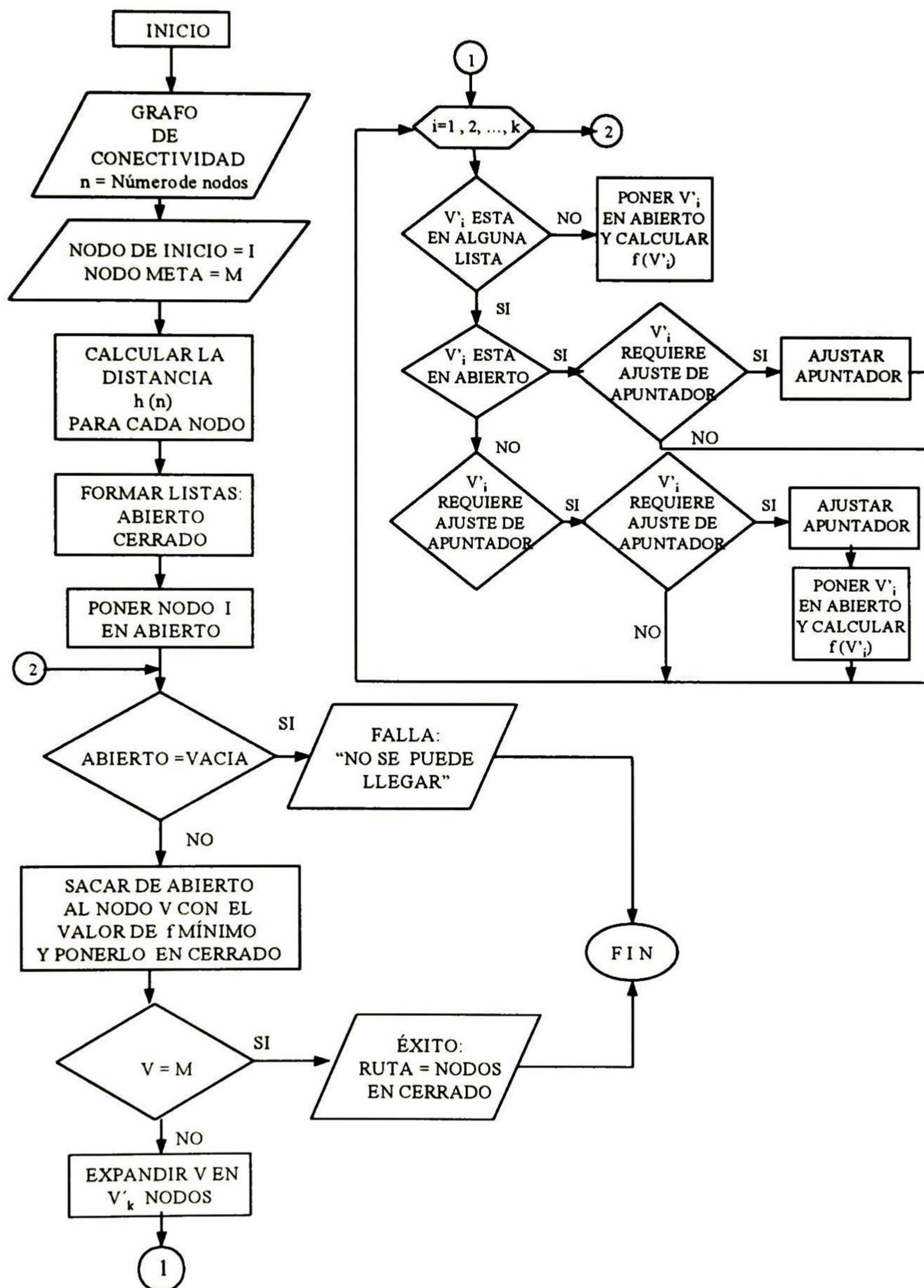


Figura 6.2: Algoritmo A*.

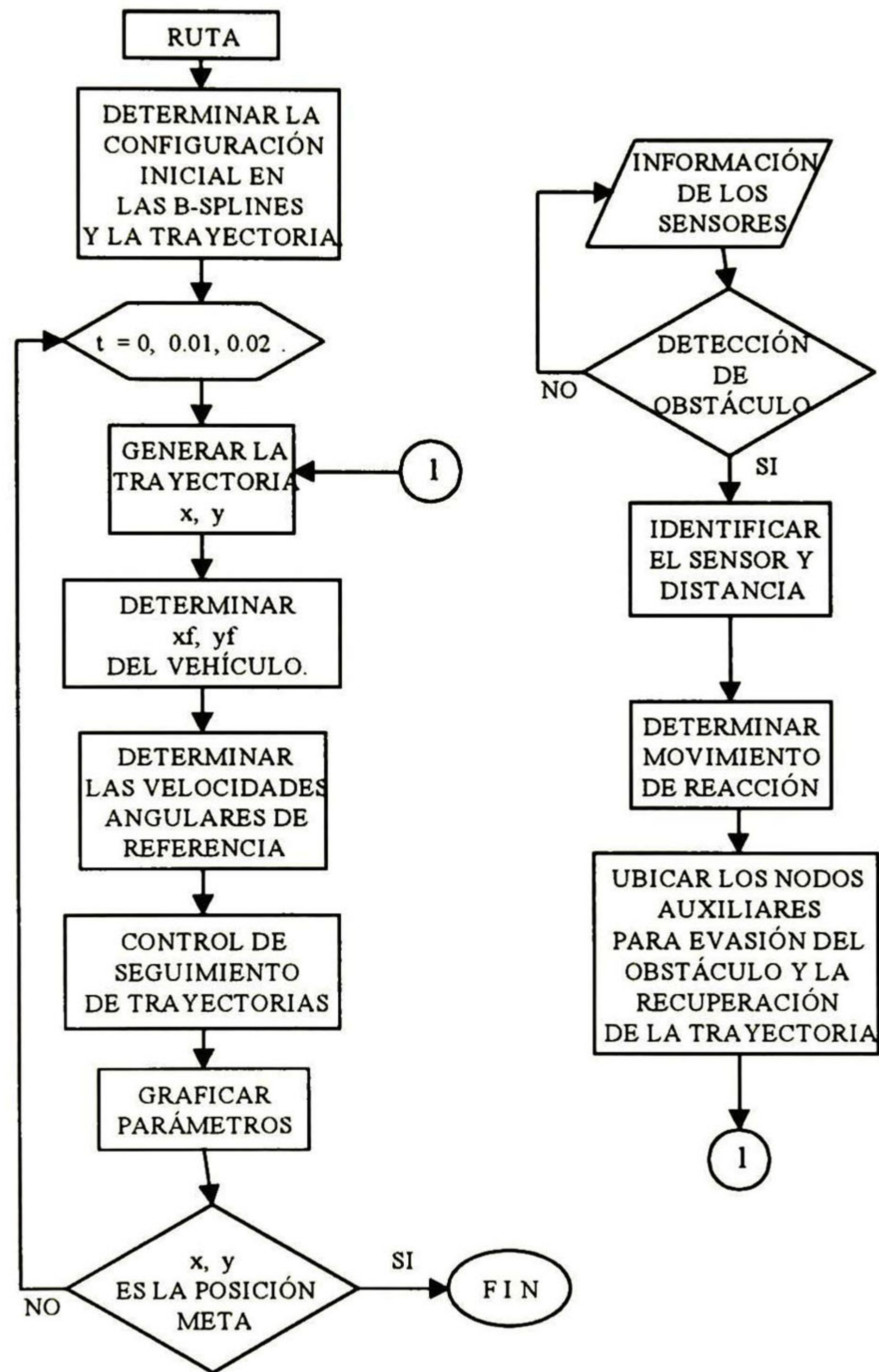


Figura 6.3: Algoritmo de generación y seguimiento de la trayectoria, en operación paralela con el algoritmo de tareas reactivas.

otros algoritmos en él.

APENDICE 2. INSTRUCTIVO DE OPERACION DEL PROGRAMA SIMULADOR.

El presente instructivo tiene la finalidad de explicar paso a paso al usuario cada una de las opciones con que cuenta el programa de simulación.

I.- INICIO

Al ingresar al programa aparece la pantalla de "DATOS DEL MAPA" con 4 opciones;

- 1.- Introducir Vértices.
- 2.- Vértices Adyacentes.
- 3.- Dar Inicio y Meta
- 4.- Terminar Programa..

1.- Introducir Vértices. Esta opción permite introducir los vértices o nodos del grafo de conectividad en sus posiciones correspondientes (x,y).

2.- Vértices Adyacentes. Permite introducir al programa la relación de adyacencia entre los vértices dados.

3.- Dar Inicio y Meta. Aquí se asignan los vértices correspondientes al inicio y la meta en el recorrido del vehículo.

4.- Terminar Programa. Finaliza la sesión.

Introduzca el número correspondiente a la operación deseada y presione ENTER.

NOTAS:

A.- En la construcción del grafo de conectividad se deben emplear inicialmente las opciones 1 y 2, para posteriormente emplear la opción 3.

B.- Una vez construido el grafo de conectividad con 1 y 2, la opción 3 se puede repetir para diferentes puntos de inicio y meta.

1.- INTRODUCIR VERTICES.

El programa esperará que se introduzcan los vértices con sus coordenadas de posición (x,y). Primero introduzca las etiquetas de los vértices. Las etiquetas aceptadas son numéricas, así que puede dar numeración consecutiva a los nodos o vértices creados.

En **vértice:** _ introduzca la etiqueta y presione ENTER.

En **posición x:** _ introduzca la coordenada x del vértice.

En **posición y:** _ introduzca la coordenada y del vértice.

Introduzca de esta manera todos los vértices del grafo y finalice introduciendo en **vértice:** un 1 para regresar a la pantalla "DATOS DEL MAPA"

NOTA: Los límites de los ejes de coordenadas en pantalla son de:

eje x: 0 a 20 m.

eje y: -6 a +6 m.

Así que la posición x, y de los vértices debe estar dentro de los límites.

Teniendo una área de simulación de 240 m²

2.- VERTICES ADYACENTES.

El programa recibirá la relación de adyacencia entre los vértices creados. Para establecer esta relación introduzca un vértice cualquiera del grafo y enseguida un vértice adyacente a él.

NOTA: Al establecer una relación de adyacencia no importa el orden en que se introducen los vértices ya que se establece la relación del primero con el segundo y del segundo con el primero.

Para finalizar, introduzca un 0 en **vértice:** y regresar a la pantalla "DATOS DEL MAPA".

3.- DAR INICIO Y META.

Se introducen las etiquetas de los vértices del grafo de inicio y meta respectivamente.

Al introducir los vértices de inicio y meta el algoritmo A* determina la ruta óptima entre estos vértices.

Los datos de longitud de la ruta óptima, y los vértices que la forman es desplegada en pantalla.

Al presionar cualquier tecla aparecerá en pantalla :

SIMULACION DE SEGMENTOS DE TRAYECTORIA. Con un eje de coordenadas donde se grafican los vértices del grafo que forman la ruta determinada por A*, que formarán los puntos de control en la trayectoria descrita por las B-splines. También se muestra el polígono de control formado por los puntos de control de la trayectoria.

Si se presiona cualquier tecla comenzará la simulación del seguimiento de la trayectoria.

En la pantalla del *Simulador del Seguimiento de Trayectoria* se despliegan los datos siguientes:

wrd: Es la velocidad angular de referencia para el motor de la rueda derecha.

wri: Es la velocidad angular de referencia para el motor de la rueda izquierda.

wd: Es la velocidad angular del motor de la rueda derecha.

wi: Es la velocidad angular del motor de la rueda izquierda.

fi: Es el ángulo de la curva de referencia.

psi: Es el ángulo descrito por la trayectoria real de seguimiento del vehículo.

wrd_max: Es la velocidad angular máxima del motor de la rueda derecha en la trayectoria.

wri_max: Es la velocidad angular máxima del motor de la rueda izquierda en la trayectoria.

x: Es la coordenada en el eje x de la trayectoria de referencia.

y: Es la coordenada en el eje y de la trayectoria de referencia.

xf: Es la coordenada en el eje x de la trayectoria del vehículo.

yf: Es la coordenada en el eje y de la trayectoria del vehículo.

vel: Es la velocidad del vehículo.

t: Es el tiempo transcurrido.

dS1: Es la distancia a la que es detectado un obstáculo por el sensor S1.

dS2: Es la distancia a la que es detectado un obstáculo por el sensor S2.

dS3: Es la distancia a la que es detectado un obstáculo por el sensor S3.

Durante la simulación aparece un obstáculo que se puede mover en la pantalla presionando las teclas de cursor.

Al terminar la simulación presione cualquier tecla para regresar a la pantalla de "DATOS DEL MAPA" donde podrá seleccionar con el número 3 nuevos vértices de inicio y meta; modificar el grafo de conectividad con 1 y 2 o terminar el programa con 4.

NOTA: En la pantalla de "DATOS DEL MAPA", se puede modificar la relación de adyacencia sin modificar la posición de los vértices creados.

Si se reubica uno o más vértices y se mantiene la relación de adyacencia con los demás, debe introducir todos los vértices con sus posiciones sin alterar la red de adyacencia.

Bibliografía

- [1] Anderson, T.L., Donath, M., "Synthesis of Reflexive Behavior for a Mobile Robot Based Upon a Stimulus-Response Paradigm", SPIE Mobile Robots III, Vol. 1007, W. Wolfe, Ed., Cambridge, MA, pp. 198-211, Nov 1988.
- [2] Arkin, R.C., "Motor-Schema-Based Navigation for a Mobile Robot: and Approach to Programming by Behavior", IEEE International Conference on Robotics and Automation, Raleigh, NC, 1987.
- [3] Arkin, R.C., "Motor-Schema-Based Mobile Robot Navigation", International Journal of Robotics Research, Vol. 8, No. 4, pp. 92-112, Ago 1989.
- [4] Arkin, R.C., Murphy, R.R., "Autonomous Navigation in a Manufacturing Environment", IEEE Transactions on Robotics and Automation, Vol. 6, No. 4, pp. 445-454, Ago 1990.
- [5] Benchimol, Guy, and Levine, Pierre, "Los Sistemas Expertos en la Empresa", Ed. Macrobot Corporation 1990.
- [6] Brooks, R.A., "A Robust Layered Control System for a Mobile Robot", IEEE Journal of Robotics and Automation, Vol. RA-2, No. 1, pp. 14-20, 1986.
- [7] Everett, H.R., "A Microprocessor Controlled Autonomous Sentry Robot", Masters Thesis, Naval Postgraduate School, Monterey, CA., Oct 1982.
- [8] Everett, H.R., "Survey of Collision Avoidance and Ranging Sensors for Mobile Robots", Technical Report 1194, Naval Command Control and Ocean Surveillance Center, San Diego, CA., Dic 1992.
- [9] Everett, H.R., "Sensors for Mobile Robots: Theory and Application", Ed. A.K. Peters, Ltd., 1995.
- [10] Gary, Chartrand, and Oellermann, Ortrud R., "Applied and Algorithmic Graph Theory", Ed. Mc Graw Hill, 1993.
- [11] Groover, Mikell P., "Automation, Production Systems, and Computer Integrated Manufacturing", Ed. Prentice Hall, 1987.

- [12] Krogh, B.H., Thorpe, C.E., "Integrated Path Planning and Dynamic Steering Control for Autonomous Vehicles", IEEE International Conference on Robotics and Automation, San Francisco, CA., pp. 1664-1669, Abr 1986.
- [13] Latombe, Jean Claude, "Robot Motion Planning", Ed. Kluwer Academic Publishers, 1996.
- [14] Luger, George F., and Stubblefield, William A., "Artificial Intelligence: Structures and Strategies for Complex Problem Solving", Ed. The Benjamin/Cummings Publishing Company, Inc., 1993.
- [15] Luggen, William W., "Flexible Manufacturing Cells and Systems", Ed. Prentice Hall, 1991.
- [16] Mathews, John H., "Numerical Methods for Mathematics, Science, and Engineering", Ed. Prentice Hall, 1992.
- [17] McQuistan, R. B., "Campos Escalares y Vectoriales", Ed. Limusa, México, 1985.
- [18] Pallas-Areny, R., and Webster, J.G., "Ultrasonic Based Sensors", Sensors, pp.16-20, Jun 1992.
- [19] Russell, Stuart, and Norving, Peter, "Artificial Intelligence: A Modern Approach", Ed. Prentice Hall, 1995.
- [20] Schwartz, J.T., and Yap, C.K., "Algorithmic and Geometric Aspects of Robotics", Ed. Lawrence Erlbaum Associates, 1987.
- [21] Swokowski, Earl W., "Algebra y Trigonometría con Geometría Analítica", Ed. Grupo Editorial Iberoamérica, 1987.
- [22] Symon, K.R. "Mecánica", Ed. Aguilar, España 1977.
- [23] Thorpe, C.E., "FIDO: Vision and Navigation for a Mobile Robot", PhD Thesis, Carnegie Mellon University, Pittsburg, PA, Nov 1984.
- [24] Thorpe, C.E., "Path Relaxation: Path Planning for a Mobile Robot", Proceedings, AAAI-84, National Conference on Artificial Intelligence, University of Texas, Austin, TX, Ago 1984.
- [25] Vázquez, Benjamín, "Uso de los B-splines en el Modelado de Curvas: Aplicaciones al Control de un Robot Móvil", Tesis de Maestría, CINVESTAV, México, 1994.
- [26] Yamaguchi, F., "Curves and Surfaces in Computer Aided Geometry Design", Ed. Springer Verlag, 1988.
- [27] Zill, Dennis G., "Cálculo con Geometría Analítica", Ed. Grupo Editorial Ibaroamérica, 1985.



**CENTRO DE INVESTIGACION Y DE ESTUDIOS AVANZADOS DEL IPN
UNIDAD GUADALAJARA**

El Jurado designado por el Laboratorio de Ingeniería Eléctrica y Ciencias de la Computación del Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, aprobó la tesis: "Simulador de un Vehículo Autoguiado" el día 28 de agosto de 1998.

Dr. Edgar Sánchez Camperos
Profesor Investigador 3B
CINVESTAV DEL IPN
Guadalajara.

Dra. Ofelia Begovich Mendoza
Profesor Investigador 3A
CINVESTAV DEL IPN
Guadalajara.

Dr. Luis Ernesto López Mellado
Profesor Investigador 3A
CINVESTAV DEL IPN
Guadalajara.

Dr. Antonio Ramírez Treviño
Profesor Investigador 2A
CINVESTAV DEL IPN
Guadalajara.

Dr. José Javier Ruiz León
Profesor Investigador 2A
CINVESTAV DEL IPN
Guadalajara.



CINVESTAV
BIBLIOTECA CENTRAL



SSIT000003819