

V 1.4

MANUAL DE AYUDA

José Luis Guzmán
Oliver López
Manuel Berenguel
Francisco Rodríguez
Sebastián Dormido

Referencia al internal report:

MRIT – Mobile Robotics Interactive Tool.

© José Luis Guzmán*, Oliver López*, Manuel Berenguel*, Francisco Rodríguez*, Sebastián Dormido**

* Universidad de Almería, Dpto. de Lenguajes y Computación, Área de Ingeniería de Sistemas y Automática. Ctra. Sacramento s/n, 04120, Almería, España.
Tel. +34 950 015683, Fax. +34 950 015129,
E-mail: [joguzman, olopez, beren, frrodrig]@ual.es.

** Universidad Nacional de Educación a Distancia, Dpto. de Informática y Automática.
Avda. Senda del Rey nº 7, 28040, Madrid, España.
Tel. +34 91 3987151, Fax. +34 91 3986697,
E-mail: sdormido@dia.uned.es

Índice

1. Introducción.....	1
2. Herramienta Interactiva de Robótica Móvil MRIT.....	3
2.1. Descripción de la herramienta.....	3
2.2. Menú de opciones.....	4
2.3. Entorno de trabajo.....	6
2.3.1. Robot móvil.....	6
2.3.2. Obstáculos.....	7
2.4. Algoritmos.....	12
2.4.1. Algoritmos Globales.....	12
2.4.2. Algoritmos Reactivos.....	17
2.5. Cinemáticas.....	21
2.6. Simulación.....	23
2.6.1. Simulación Paso a Paso.....	24
2.6.2. Simulación en modo Ejecutar.....	28
2.6.3. Mensajes de Error.....	28
3. Ejemplos.....	31
3.1. Análisis y comparación de algoritmos de navegación.....	31
3.1.1 Algoritmos locales.....	31
3.1.1.1. BUG.....	31
3.1.1.2. VISBUG.....	31
3.1.1.3. Potencial.....	34
3.1.2. Algoritmos globales.....	36
3.1.2.1. Grafo de visibilidad.....	36
3.1.2.2. Diagrama de Voronoi.....	37
3.1.2.3. Quadtree.....	37
3.1.2.4. Frente de onda.....	39
3.1.3. Comparación de algoritmos de navegación.....	41
3.1.4. Estudio de persecución pura.....	43
3.2. Comparación de comportamiento de cinemáticas.....	43
3.2.1. Ejemplos de las distintas cinemáticas.....	45
3.2.2. Estudio de restricciones.....	47

Índice de figuras

2.1. Pantalla principal de la herramienta interactiva.....	3
2.2. Menús de selección del modo de funcionamiento.....	4
2.3. Menús de selección de algoritmo.....	4
2.4. Menús de selección de los parámetros del algoritmo seleccionado.....	5
2.5. Menús de selección de los parámetros del robot móvil.....	5
2.6. Menús de selección de las cinemáticas del robot móvil.....	6
2.7. Menús de selección de las operaciones con obstáculos.....	6
2.8. Cambio del tamaño y orientación del robot móvil.....	7
2.9. Ejemplo de cambio de la orientación inicial.....	7
2.10. Ejemplo de cambio de las posiciones inicial y final del robot móvil.....	8
2.11. Menús de selección del tipo de obstáculo a insertar.....	8
2.12. Ejemplo de selección de obstáculos triángulo y rectángulo.....	9
2.13. Ejemplo de selección de los distintos tipos de obstáculos polígono disponibles.....	9
2.14. Ejemplo de selección del tipo de obstáculo polígono a insertar.....	10
2.15. Ejemplo de entorno de trabajo con un obstáculo de cada tipo.....	10
2.16. Menús de selección para mover un obstáculo del entorno de trabajo.....	11
2.17. Ejemplo de modificación de un obstáculo del entorno de trabajo.....	11
2.18. Ejemplo de fusión de dos obstáculos del entorno de trabajo.....	11
2.19. Menús de selección para eliminar un obstáculo del entorno de trabajo.....	12
2.20. Ejemplo de eliminación de un obstáculo del entorno de trabajo.....	12
2.21. Menús y ejemplos de selección de los parámetros del algoritmo Persecución Pura..	13
2.22. Ejemplo del efecto del aumento del tamaño del robot móvil en el entorno de trabajo.....	14
2.23. Menús y ejemplo de selección del parámetro ‘Área Quadtree’.....	15
2.24. Ejemplo del efecto de modificar el parámetro ‘Área Quadtree’.....	15
2.25. Menús y ejemplo de selección del parámetro ‘tamaño de Celda’.....	16
2.26. Ejemplo del efecto de modificar los parámetros ‘tamaño de Celda’ y ‘vecindad’...	16
2.27. Menús de selección del parámetro ‘sentido del giro’ en el algoritmo BUG.....	17
2.28. Ejemplo del efecto de modificar el parámetro ‘sentido del giro’ en el algoritmo BUG.....	17
2.29. Ejemplos de varios valores para el parámetro ‘alcance del sónar’.....	18
2.30. Menús y ejemplo de selección del parámetro ‘alcance del sónar’.....	19
2.31. Ejemplo del efecto de modificar los parámetros ‘sentido de giro’ y ‘alcance del sónar’.....	19
2.32. Menús y ejemplo de selección de los parámetros ‘alcance del sónar’, ‘atracción’ y ‘repulsión’.....	20
2.33. Ejemplo de los efectos de modificar los parámetros ‘alcance del sónar’, ‘atracción’ y ‘repulsión’.....	21
2.34. Menús de selección de la cinemática ‘diferencial’ y efecto sobre la interfaz de usuario.....	21
2.35. Menús de selección de la cinemática ‘triciclo’ y efecto sobre la interfaz de usuario	22
2.36. Menús de selección de la cinemática ‘síncrona’ y efecto sobre la interfaz de usuario.....	22
2.37. Ejemplos del efecto de aplicar restricciones a las diferentes cinemáticas.....	23
2.38. Menús de selección de los modos de ejecución de la simulación.....	24
2.39. Menús y ejemplo de selección del parámetro ‘paso actual’.....	24

2.40. Pasos de simulación del algoritmo ‘grafo de visibilidad’	25
2.41. Pasos de simulación del algoritmo ‘diagrama de Voronoi’	26
2.42. Pasos de simulación del algoritmo ‘quadtree’	27
2.43. Pasos de simulación del algoritmo ‘frente de onda’	28
2.44. Mensaje de error al lanzar la ejecución de la simulación	29
2.45. Mensaje de error de exceso de obstáculos en el entorno de trabajo	29
2.46. Mensaje de error de colisión del robot móvil	29
2.47. Mensaje de error de mínimo local en el entorno de trabajo	29
3.1. Algoritmo BUG con tamaño del robot 2	32
3.2. Algoritmo BUG con tamaño del robot 3	32
3.3. Algoritmo BUG con tamaño del robot 4	32
3.4. Algoritmo VISBUG con alcance del s3onar 3 y tama1o del robot 2	33
3.5. Algoritmo VISBUG con alcance del s3onar 6 y tama1o del robot 3	33
3.6. Algoritmo VISBUG con alcance del s3onar 8 y tama1o del robot 4	33
3.7. Algoritmo Potencial con alcance del s3onar 4	34
3.8. Algoritmo Potencial con atracci3n 2 y repulsi3n 1	35
3.9. Algoritmo Potencial con alcance del s3onar 6, atracci3n 5, repulsi3n 2 y tama1o 4	35
3.10. Algoritmo grafo de visibilidad con distintos tama1os de robot	36
3.11. Fusi3n de obst3culos con algoritmo grafo de visibilidad	37
3.12. Algoritmo Diagrama de Voronoi con distintos tama1os de robot	38
3.13. Algoritmo quadtree con distintas 3reas quadtree y tama1o de robot 2	38
3.14. Algoritmo quadtree con distintas 3reas quadtree y tama1o de robot 3	39
3.15. Algoritmo quadtree con distintas 3reas quadtree y tama1o de robot 4	39
3.16. Algoritmo frente de onda con tama1o de celda 4 y tama1o de robot 2	40
3.17. Algoritmo frente de onda con tama1o de celda 5 y tama1o de robot 2	40
3.18. Algoritmo frente de onda con tama1o de celda 6 y tama1o de robot 2	40
3.19. Ejemplo de los algoritmos de planificaci3n local	41
3.20. Ejemplo de los algoritmos de planificaci3n global	42
3.21. Ejemplo de persecuci3n pura con distintos valores de tiempo de muestreo T_m	43
3.22. Ejemplo de persecuci3n pura con distintos valores de distancia al punto objetivo d	44
3.23. Ejemplo de persecuci3n pura con distinta orientaci3n inicial	44
3.24. Ejemplo de persecuci3n pura con distinta velocidad y la misma orientaci3n inicial	45
3.25. Algoritmo potencial con distintas cinem3ticas	45
3.26. Algoritmo quadtree con distintas cinem3ticas	46
3.27. Se1ales de control caracter3sticas de las distintas cinem3ticas	47
3.28. Ejemplo de cinem3tica diferencial con y sin restricciones	48
3.29. Se1ales de control caracter3sticas de la cinem3tica diferencial con y sin restricciones	48
3.30. Ejemplo de cinem3tica triciclo con restricciones	49
3.31. Se1al de control caracter3stica de la cinem3tica triciclo con restricciones	49
3.32. Ejemplo de cinem3tica s3ncrona con restricciones	50
3.33. Se1al de control caracter3stica de la cinem3tica s3ncrona con restricciones	50

1. Introducción.

La herramienta interactiva *MRIT (Mobile Robotics Interactive Tool; Herramienta Interactiva para Robótica Móvil)* se utiliza para la prueba y diseño y prueba de algoritmos de navegación para robótica móvil. Es una herramienta con fines de investigación y docencia la cual sirve tanto para asentar conceptos teóricos como para investigar nuevas técnicas y métodos para robótica móvil. Dado que es una herramienta interactiva, facilita mucho su uso y la rápida experimentación por parte del usuario.

Desde el punto de vista docente, el alumno puede poner en práctica los conocimientos adquiridos en las asignaturas mediante sencillos ejemplos simulados. La herramienta ayuda al alumno a comprender la teoría subyacente bajo los algoritmos de planificación y seguimiento de trayectorias, los modelos cinemáticos y la evitación de obstáculos. Gracias a la interactividad de la herramienta, es posible observar como la respuesta de la simulación se ve afectada debido a los cambios en los valores de los distintos parámetros, tales como los propios del algoritmo seleccionado, las señales de control cinemáticas, los parámetros propios del robot móvil, etc.

Este documento es un tutorial de la herramienta, y en su segundo capítulo presenta una descripción sobre las partes de la misma y su funcionamiento. Una vez que el usuario se familiarice con ésta, en el tercer y último capítulo, se muestran un conjunto de ejemplos presentados con el objetivo de hacer comprender y poner en práctica conocimientos sobre robótica móvil al usuario.

2. MRIT: Herramienta Interactiva de Robótica Móvil.

2.1. Descripción de la herramienta.

La herramienta consta de una única pantalla en la que se muestran todos los parámetros manipulables por parte del usuario así como las gráficas donde se muestran los resultados. En la figura 2.1 se muestra la pantalla principal, donde pueden distinguirse dos grandes zonas. La parte gráfica, donde se muestran todos los resultados, ocupa las partes central y derecha de la pantalla. En ella se muestran el entorno de trabajo, que contiene al robot y los diferentes obstáculos. La otra parte, situada en la parte izquierda de la pantalla, contiene un menú con todos los parámetros relativos al robot, los obstáculos, los algoritmos, y las cinemáticas.

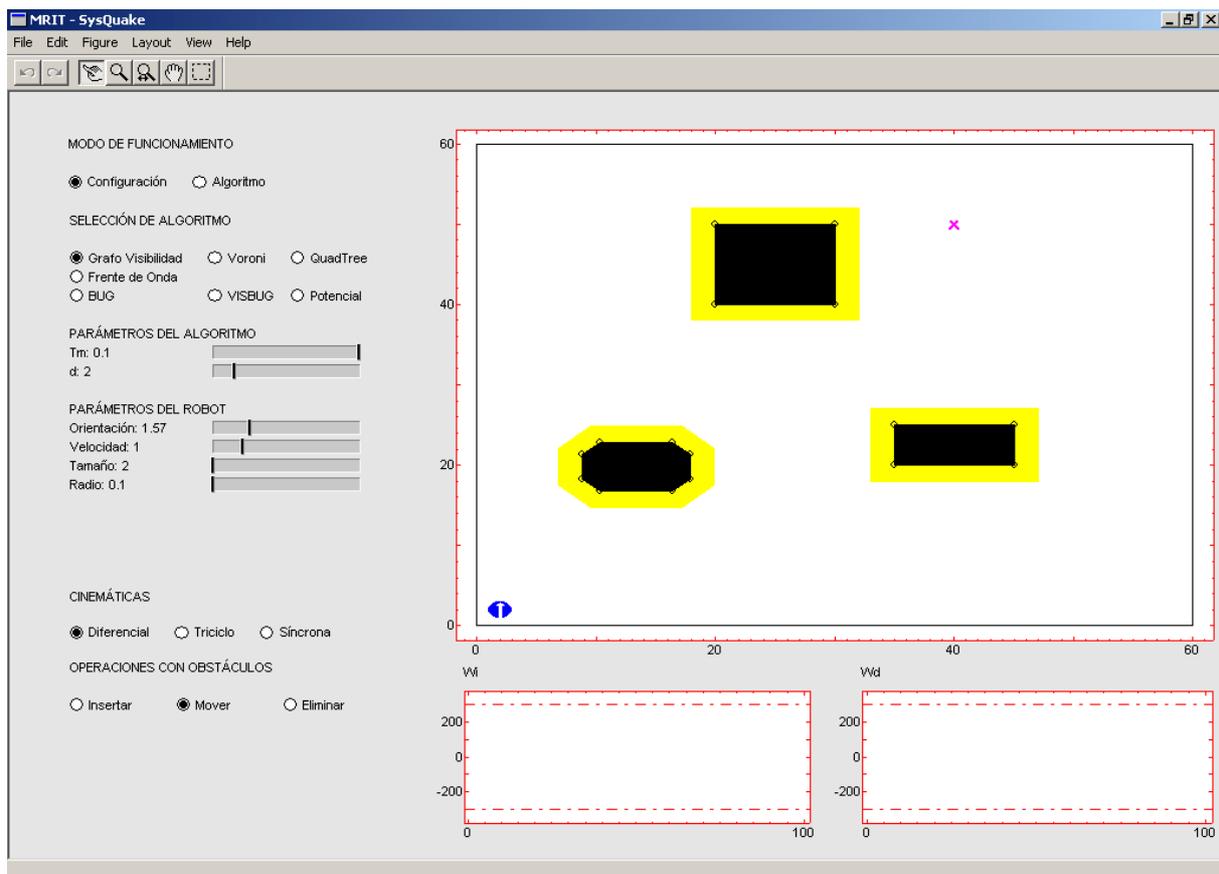


Figura 2.1. Pantalla principal de la herramienta interactiva.

2.2. Menú de opciones.

El menú de opciones se encuentra situado en la parte izquierda de la herramienta. Por motivos de consistencia de la interfaz, las opciones se encuentran duplicadas, ya que están disponibles al mismo tiempo a través del menú *Settings*, situado en la barra de menú superior, y en la parte izquierda de la herramienta. Dicho menú se encuentra dividido en varias partes:

- **Modo de funcionamiento**

La herramienta tiene dos modos de funcionamiento: *Configuración* y *Algoritmo*. El primer modo permite seleccionar el algoritmo, sus parámetros, los parámetros del robot, su cinemática, y manipular los obstáculos del entorno de trabajo. Aunque los dos modos de funcionamiento permiten la manipulación de los mismos parámetros, la principal diferencia es que en modo *Algoritmo* se ejecuta el algoritmo seleccionado junto con los parámetros correspondientes, mientras que en modo *Configuración* no se lleva a cabo ningún cálculo. Así, si estamos en modo *Algoritmo* y modificamos algún parámetro, el cambio se verá reflejado en el entorno de trabajo de forma inmediata.

En la figura 2.2 pueden observarse los menús de selección del modo de funcionamiento.

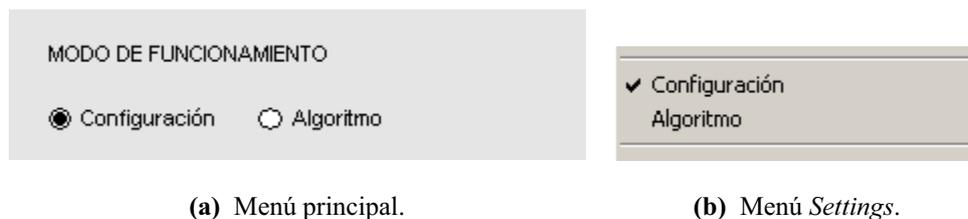


Figura 2.2. Menús de selección del modo de funcionamiento.

- **Selección de Algoritmo**

Los algoritmos que incluye la herramienta se dividen en algoritmos globales y algoritmos reactivos. Los primeros incluyen: *Grafo de Visibilidad*, *Diagrama de Voronoi*, *Quadtree* y *Frente de Onda*. Los algoritmos reactivos son: *BUG*, *VISBUG* y *Potencial*.

En la figura 2.3 se muestran los menús de selección de algoritmos.

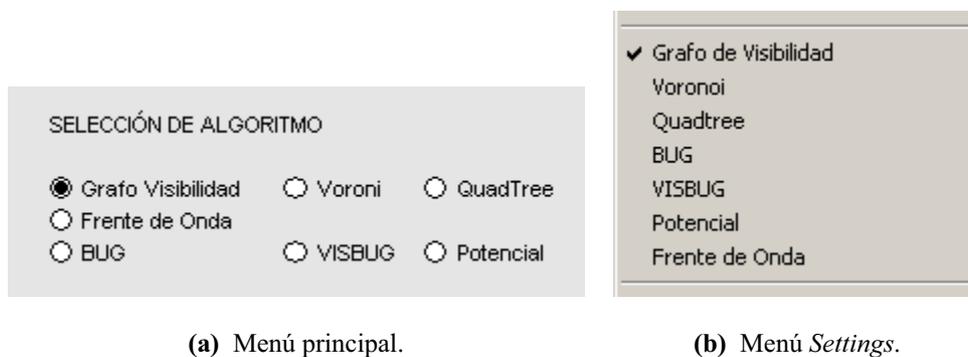


Figura 2.3. Menús de selección de algoritmo.

- **Parámetros del Algoritmo**

Los algoritmos que incluye la herramienta cuentan con una serie de parámetros característicos. En cada momento se muestran solo los parámetros específicos del algoritmo que se encuentra seleccionado.

En la figura 2.4 se muestra un ejemplo de menú de selección de los parámetros de los algoritmos. Los distintos parámetros para cada uno de los algoritmos existentes se describirá en detalle más adelante.

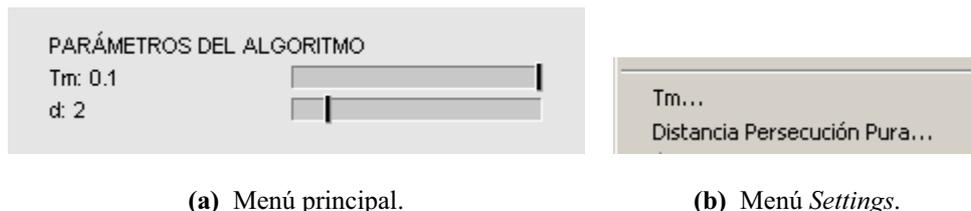


Figura 2.4. Menús de selección de los parámetros del algoritmo seleccionado.

- **Parámetros del Robot**

El robot móvil cuenta también con una serie de parámetros característicos. En este caso podemos cambiar su orientación, su velocidad, su tamaño y el radio de sus ruedas.

En la figura 2.5 se muestran los dos menús de selección de los parámetros del robot.

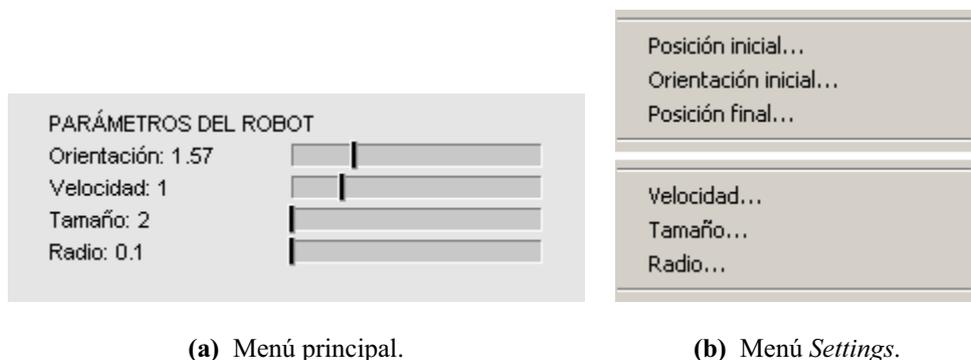


Figura 2.5. Menús de selección de los parámetros del robot móvil.

- **Cinemáticas**

Es posible seleccionar tres cinemáticas distintas para el seguimiento de trayectorias: *Diferencial*, *Triciclo* o *Ackerman* y *Síncrona*.

En la figura 2.6 se muestran los dos menús de selección de las cinemáticas del robot.

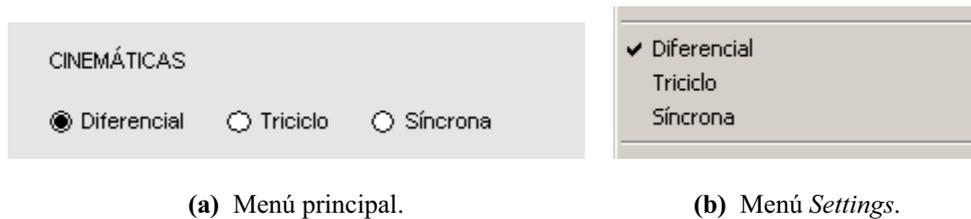


Figura 2.6. Menús de selección de las cinemáticas del robot móvil.

- **Operaciones con Obstáculos**

A la hora de definir o modificar el entorno de trabajo, el usuario puede realizar una serie de operaciones con los obstáculos del mismo: *Insertar*, *Mover* y *Eliminar*.

En la figura 2.7 se muestran los dos menús de selección de las operaciones permitidas sobre los obstáculos.

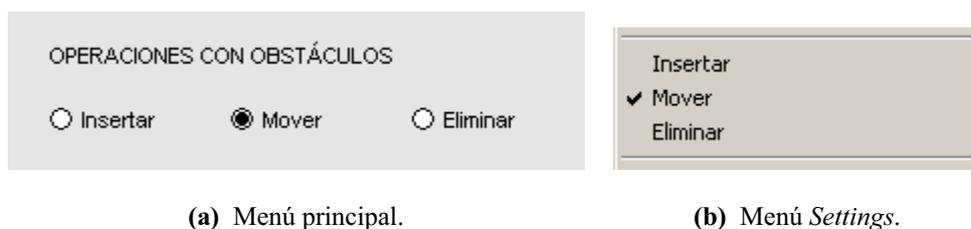


Figura 2.7. Menús de selección de las operaciones con obstáculos.

2.3. Entorno de trabajo.

El entorno de trabajo se representa mediante una gráfica que ocupa gran parte de la pantalla principal de la herramienta, en la parte central y derecha de la misma. Éste entorno bidimensional tiene unas proporciones de 6 metros cuadrados y se muestra de forma cenital, es decir, visto desde arriba. El entorno contiene al robot móvil, los obstáculos y el punto de destino.

2.3.1. Robot móvil.

El robot móvil se representa mediante un círculo azul con una flecha blanca en su interior. El círculo azul tiene un diámetro especificado por la dimensión máxima del robot y la flecha indica la orientación actual. En la figura 2.8 puede observarse un ejemplo de cambio del tamaño del robot y de su orientación.

En la figura 2.9 se muestra un ejemplo de cómo cambiar la orientación mediante el menú desplegable *Settings*.

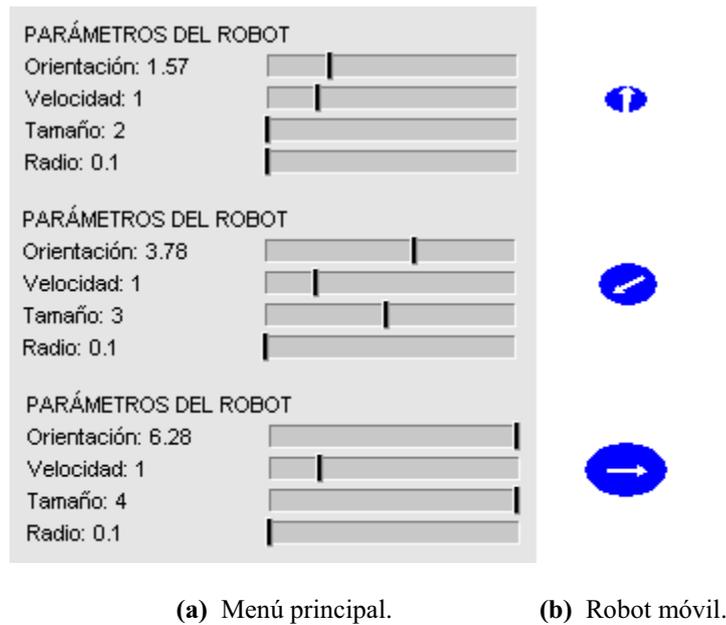


Figura 2.8. Cambio del tamaño y orientación del robot móvil.

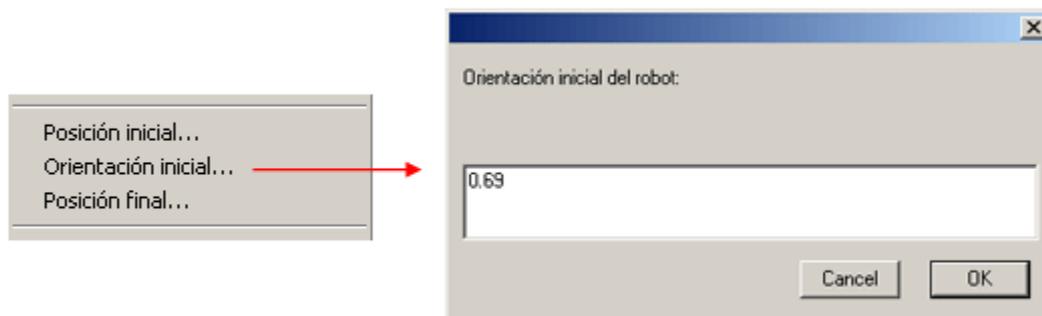
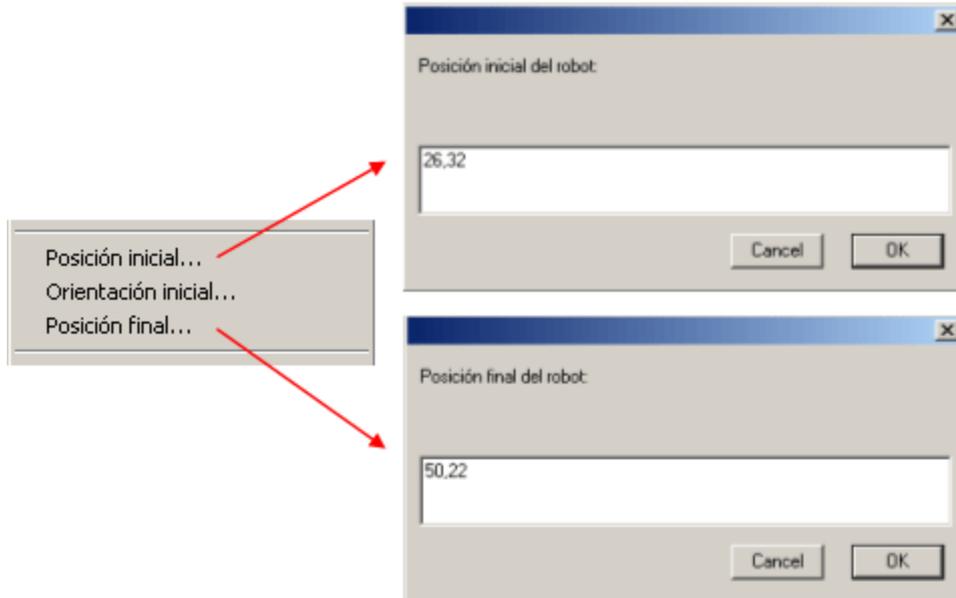


Figura 2.9. Ejemplo de cambio de la orientación inicial.

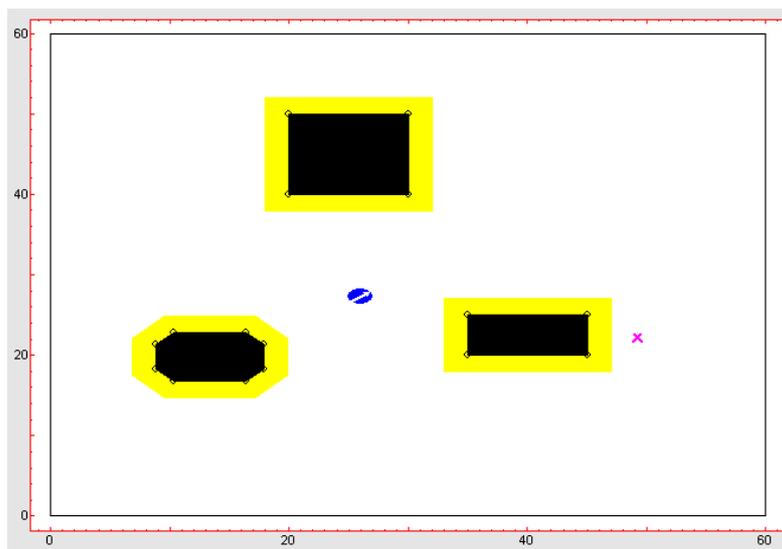
Inicialmente, el robot se encuentra situado en el punto inicial, con una orientación dada, y la meta en el punto destino, representado mediante una *X*. Ambos puntos pueden modificarse de forma interactiva arrastrándolos a cualquier otra posición libre del entorno de trabajo, y de forma exacta mediante el menú desplegable *Settings*. En la figura 2.10 (a) se muestra como pueden modificarse los puntos inicial y destino del robot móvil, y el resultado de dicha modificación (figura 2.10 (b)).

2.3.2. Obstáculos.

Los obstáculos del entorno de trabajo se representan mediante polígonos regulares. Inicialmente existen dos rectángulos y un octógono, pero puede haber hasta seis obstáculos simultáneamente en el entorno de trabajo. A la hora de insertar obstáculos se cuenta con varias formas geométricas en base al número de aristas que poseen: triángulos, rectángulos y polígonos. Éstos últimos incluyen pentágonos, hexágonos, heptágonos y octógonos. En la figura 2.11 pueden observarse los menús de selección del tipo de obstáculo a insertar.

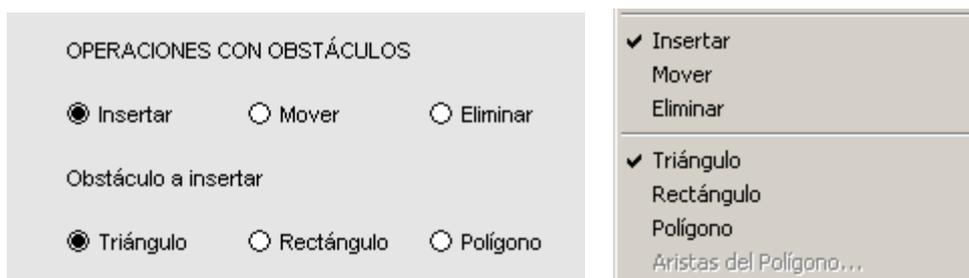


(a) Menú *Settings*.



(b) Entorno de trabajo.

Figura 2.10. Ejemplo de cambio de las posiciones inicial y final del robot móvil.



(a) Menú principal.

(b) Menú *Settings*.

Figura 2.11. Menús de selección del tipo de obstáculo a insertar.

En la figura 2.12 puede observarse un ejemplo de un triángulo y un rectángulo.

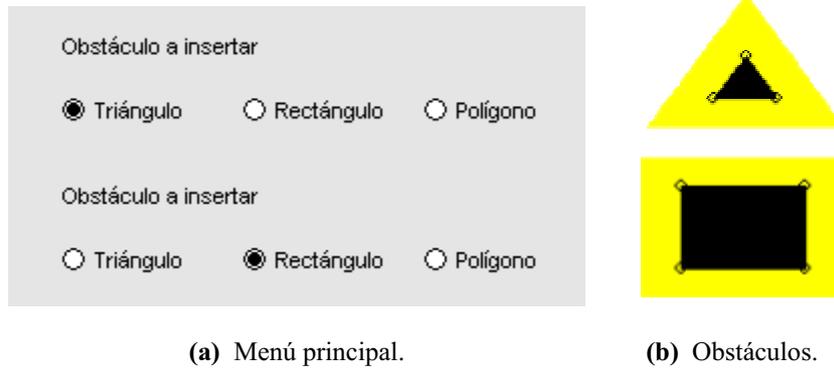


Figura 2.12. Ejemplo de selección de obstáculos triángulo y rectángulo.

En la figura 2.13 pueden observarse los menús de selección del tipo de obstáculo poligonal a insertar y el resultado de seleccionar cada uno de los polígonos disponibles.

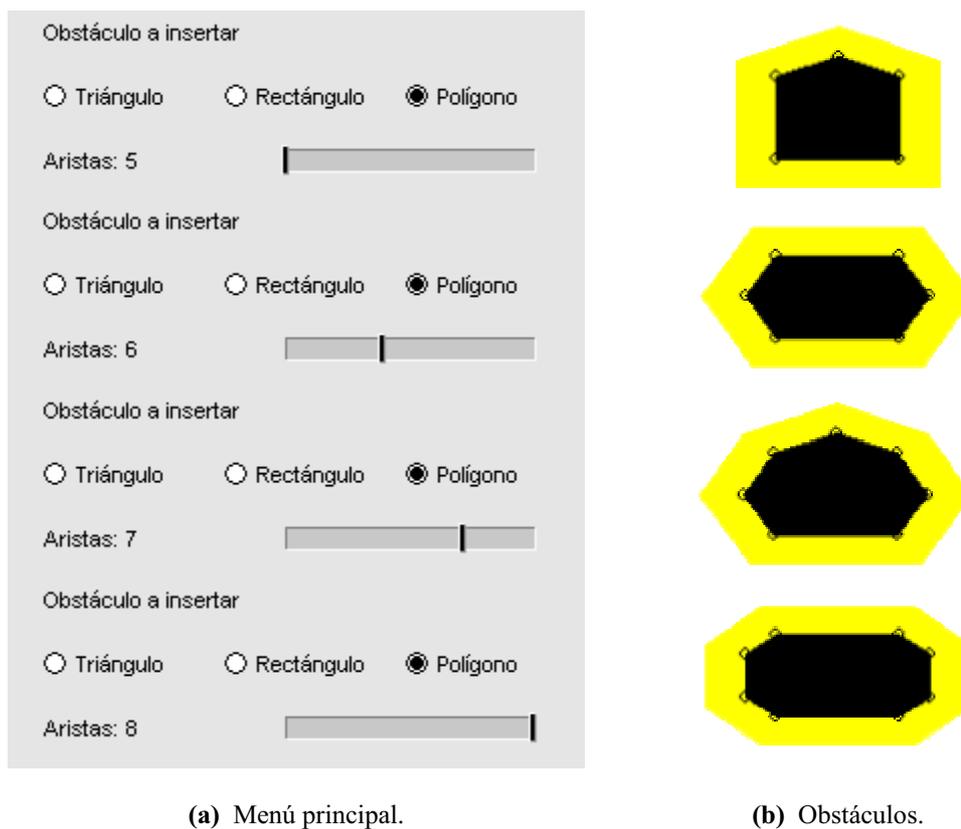


Figura 2.13. Ejemplo de selección de los distintos tipos de obstáculos polígono disponibles.

En la figura 2.14 se muestra la forma de insertar un obstáculo poligonal mediante el menú desplegable *Settings*.

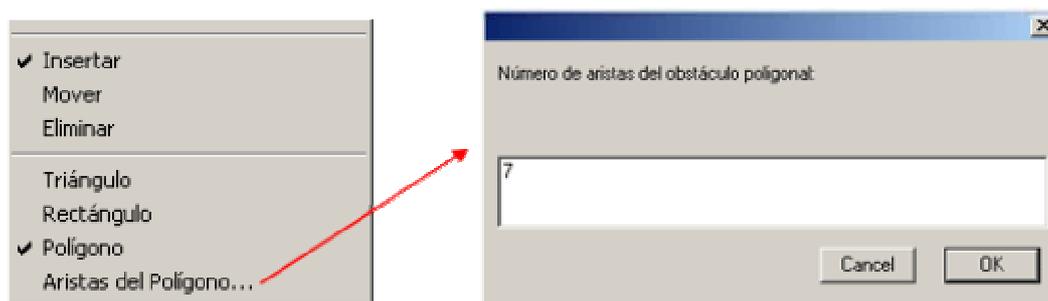


Figura 2.14. Ejemplo de selección del tipo de obstáculo polígono a insertar.

Cuando se inserta un obstáculo en el entorno de trabajo, éste se dispone alrededor del punto donde hemos indicado mediante el puntero del ratón. El tamaño de los obstáculos está predefinido de forma proporcional el tamaño del entorno de trabajo y del robot.

En la figura 2.15 puede observarse un entorno de trabajo con un obstáculo de cada tipo.

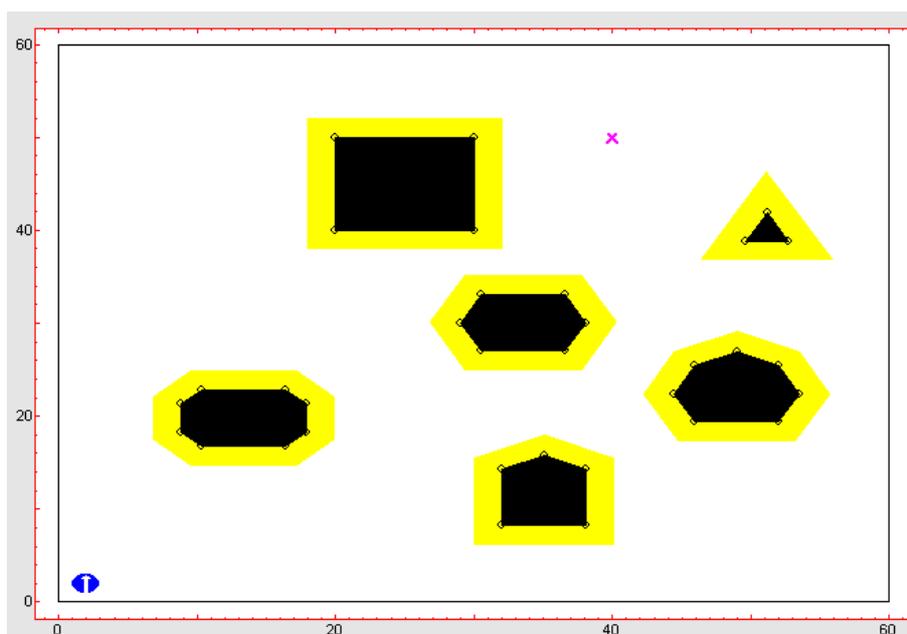


Figura 2.15. Ejemplo de entorno de trabajo con un obstáculo de cada tipo.

Los obstáculos de entorno de trabajo pueden modificar su forma mediante la opción *Mover* de los menús de opciones. Los vértices de los obstáculos son interactivos, y pueden modificar su posición, dando lugar a polígonos irregulares. En la figura 2.16 se muestran los menús de selección para mover obstáculos. En la figura 2.17 puede observarse un ejemplo de modificación de obstáculos.



(a) Menú principal.

(b) Menú *Settings*.

Figura 2.16. Menús de selección para mover un obstáculo del entorno de trabajo.

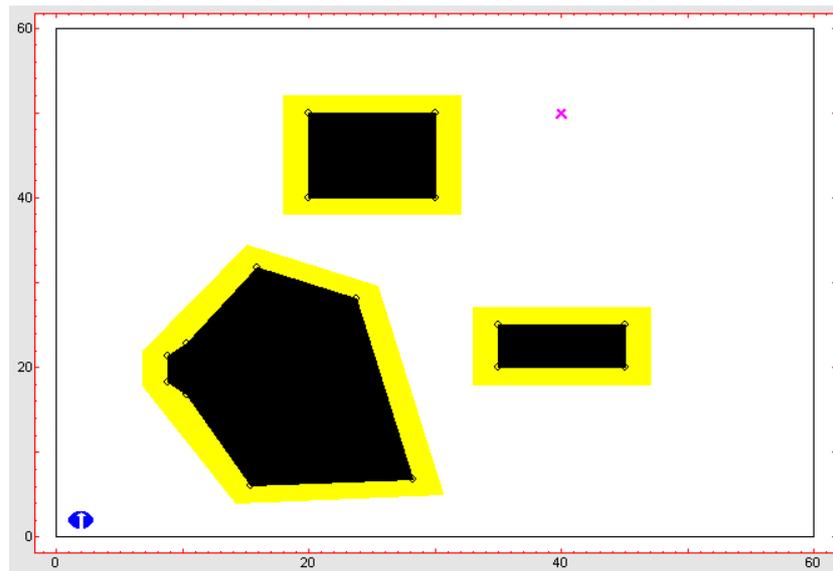


Figura 2.17. Ejemplo de modificación de un obstáculo del entorno de trabajo.

Si al modificar la forma de un obstáculo éste intersecta con uno o más obstáculos existentes, éstos se fusionan para formar un solo obstáculo (dependiendo del algoritmo seleccionado). En la figura 2.18 puede observarse un ejemplo de fusión de obstáculos.

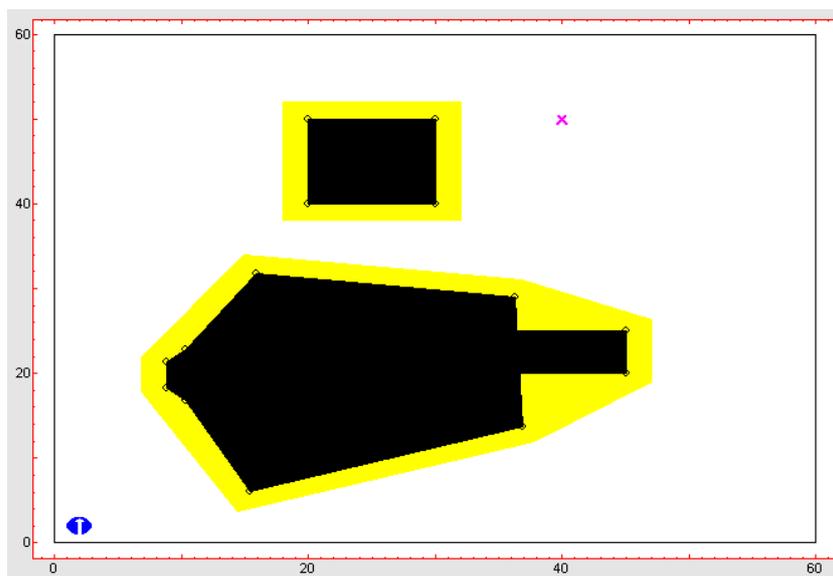


Figura 2.18. Ejemplo de fusión de dos obstáculos del entorno de trabajo.

Los obstáculos también pueden borrarse por completo, mediante la opción *Eliminar*. Seleccionando el obstáculo a borrar mediante el botón izquierdo del ratón, éste se elimina del entorno de trabajo al instante. En la figura 2.19 se muestran los menús para eliminar un obstáculo. En la figura 2.20 puede observarse el resultado de eliminar un obstáculo del entorno de trabajo mostrado en la figura 2.17.



(a) Menú principal.

(b) Menú *Settings*.

Figura 2.19. Menús de selección para eliminar un obstáculo del entorno de trabajo.

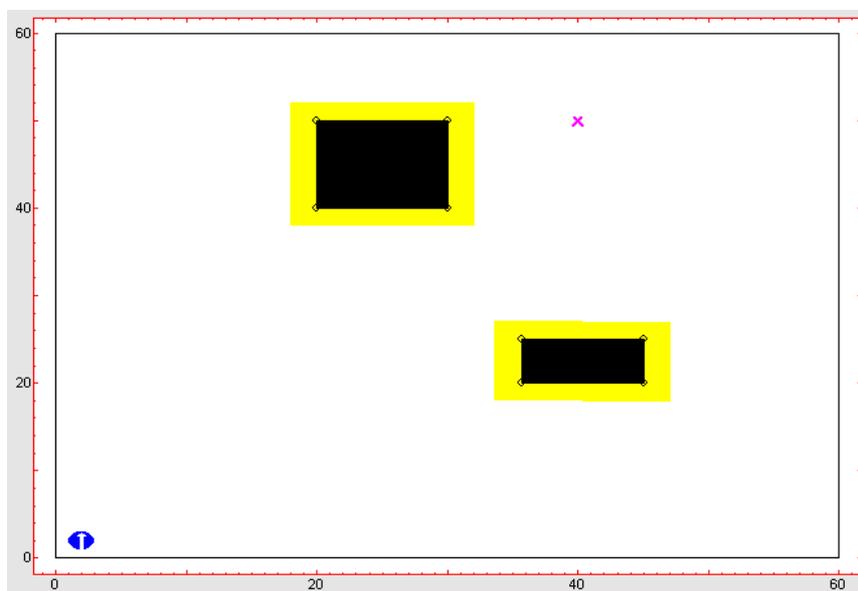
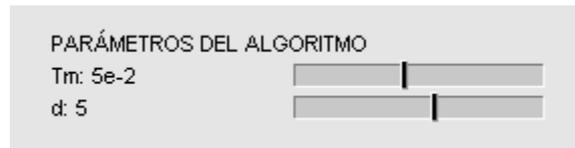


Figura 2.20. Ejemplo de eliminación de un obstáculo del entorno de trabajo.

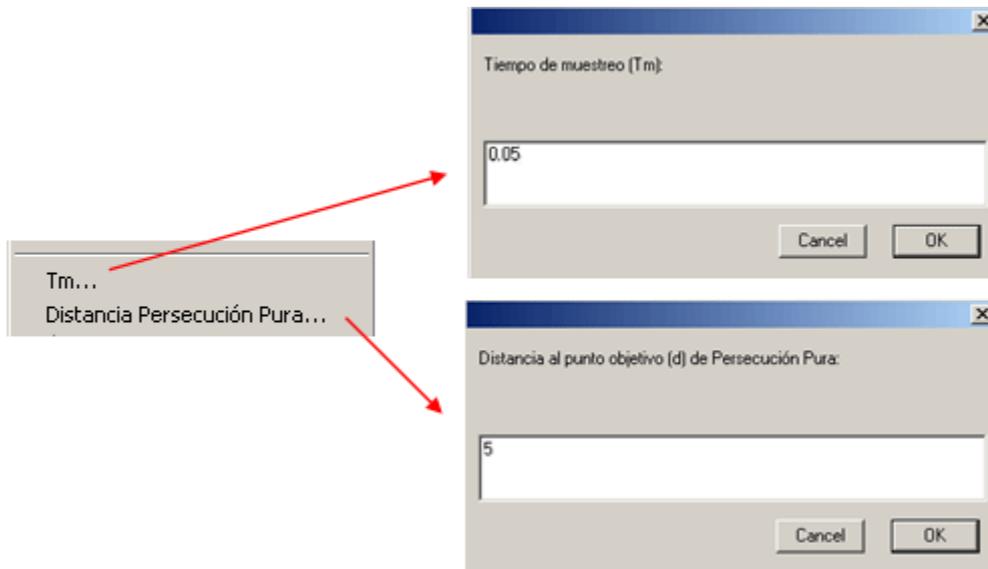
2.4. Algoritmos.

2.4.1. Algoritmos Globales.

Los algoritmos globales realizan primero la planificación de la trayectoria y después su seguimiento. El algoritmo de seguimiento que utilizan todos los algoritmos globales presentes en la herramienta es el de *Persecución Pura (Pure Pursuit)*. Así, todos ellos cuentan entre sus parámetros característicos con los dos relativos al algoritmo de Persecución Pura: el periodo de muestreo (T_m) y la distancia al punto objetivo (d). En la figura 2.21 se muestran los menús de selección y un ejemplo de selección de ambos parámetros.



(a) Menú principal.

(b) Menú *Settings*.**Figura 2.21.** Menús y ejemplos de selección de los parámetros del algoritmo Persecución Pura.

Otra característica que tienen en común los algoritmos globales es que realizan un aumento en el perímetro del obstáculo correspondiente a la dimensión máxima del robot, representada por el parámetro *Tamaño*. Esta envolvente del obstáculo tiene como objetivo el poder considerar al robot como un objeto puntual, por lo que, a mayor tamaño del robot, mayor tamaño de los obstáculos engrosados.

En la figura 2.22 se puede observar un ejemplo del efecto del aumento del tamaño del robot en los obstáculos del entorno de trabajo.

A continuación se describen los parámetros característicos que poseen algunos de los algoritmos globales.

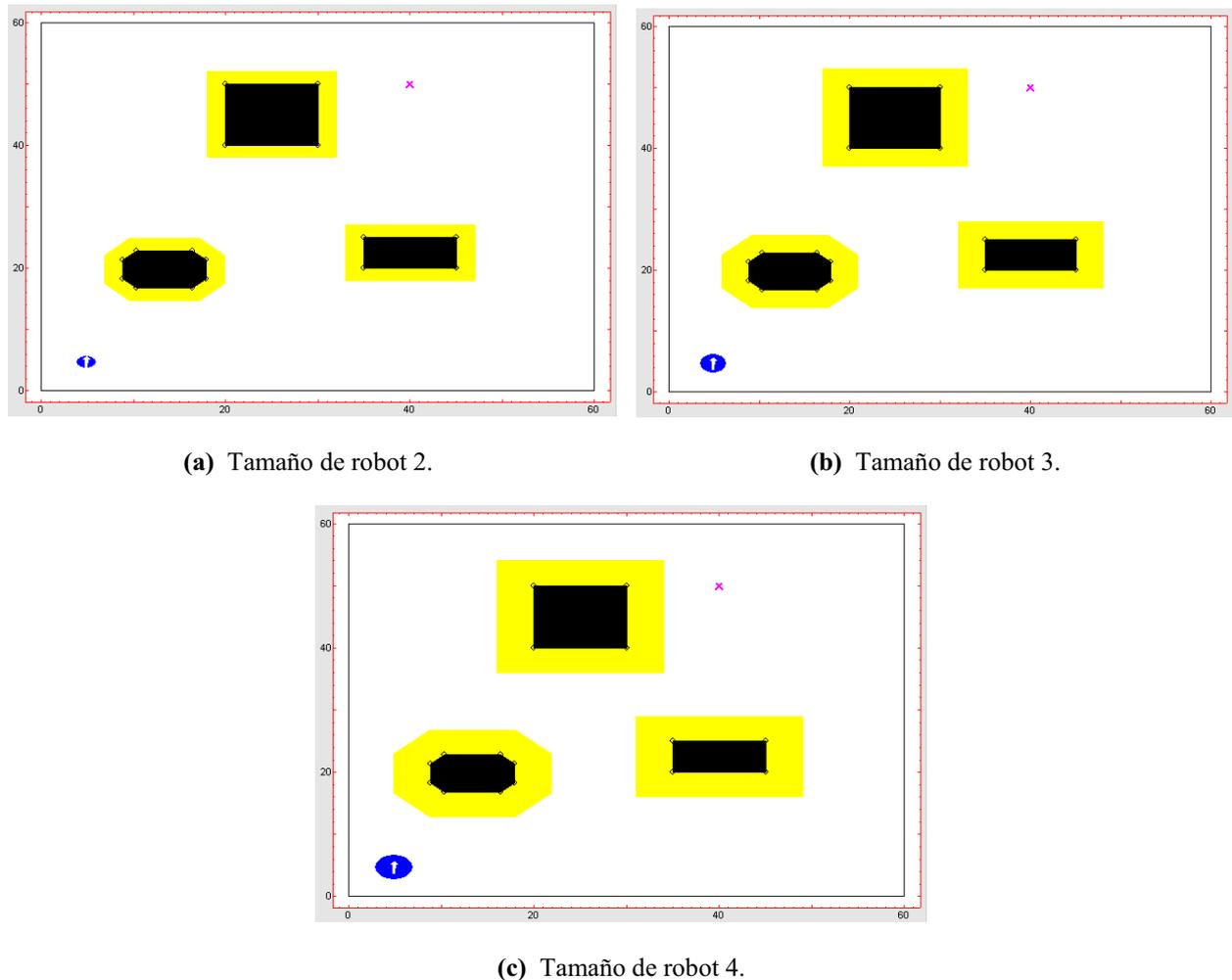


Figura 2.22. Ejemplo del efecto del aumento del tamaño del robot móvil en el entorno de trabajo.

- **Quadtree**

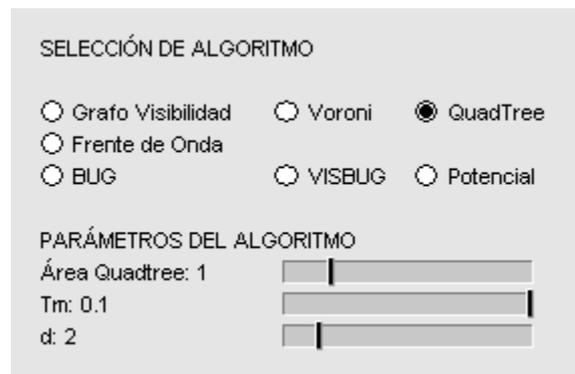
Tiene como parámetro característico el tamaño de la cuadrícula (*Área Quadtree*). En la figura 2.23 se muestran los menús de selección de dicho parámetro.

En la figura 2.24 puede observarse el efecto de modificar el tamaño de la cuadrícula sobre el algoritmo.

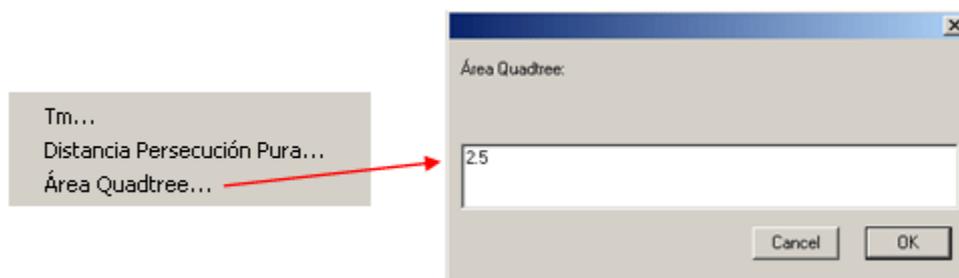
- **Frente de Onda**

Tiene como parámetros característicos el tamaño de las celdas que discretizan el entorno de trabajo y la vecindad entre estas celdas. En la figura 2.25 se muestran los menús de selección de dichos parámetros.

En la figura 2.26 puede observarse el efecto de modificar el tamaño de celda junto con la vecindad entre celdas.



(a) Menú principal.



(b) Menú *Settings*.

Figura 2.23. Menús y ejemplo de selección del parámetro 'Área Quadtree'.

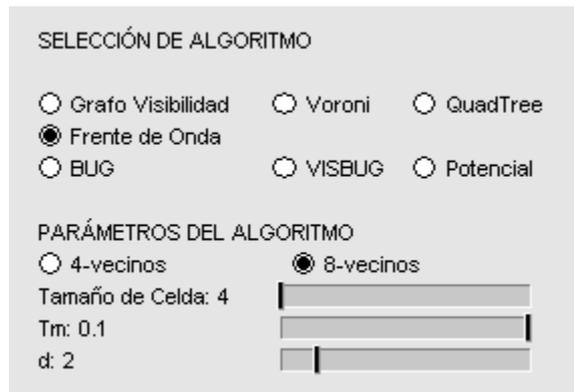


(a) Área Quadtree 1.

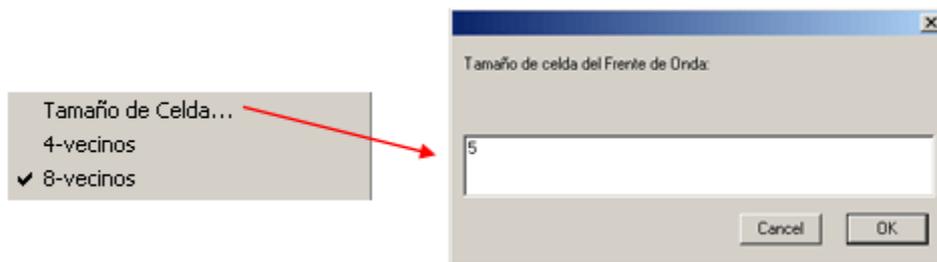


(b) Área Quadtree 2.

Figura 2.24. Ejemplo del efecto de modificar el parámetro 'Área Quadtree'.

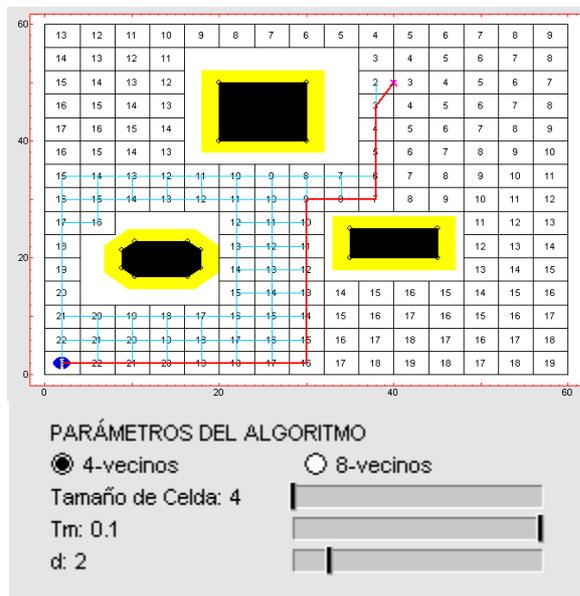


(a) Menú principal.

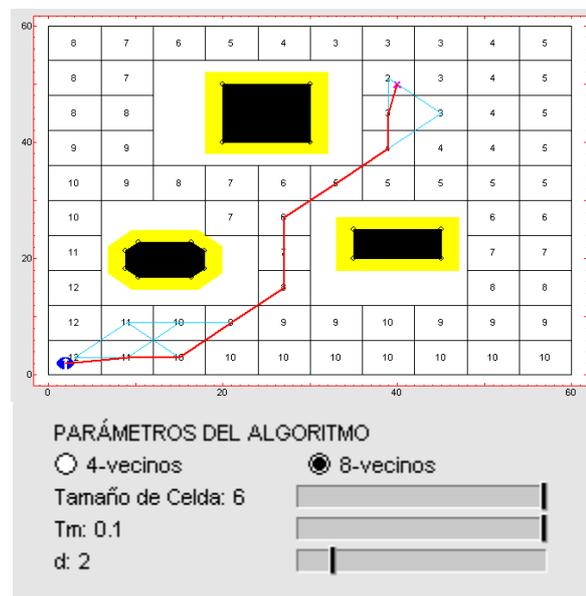


(b) Menú Settings.

Figura 2.25. Menús y ejemplo de selección del parámetro 'tamaño de Celda'.



(a) Tamaño de celda 4, vecindad 4.



(b) Tamaño de celda 6, vecindad 8.

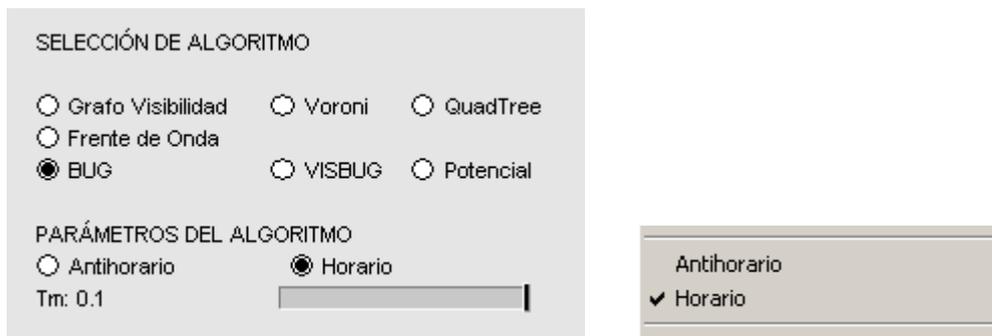
Figura 2.26. Ejemplo del efecto de modificar los parámetros 'tamaño de Celda' y 'vecindad'.

2.4.2. Algoritmos Reactivos.

Los algoritmos reactivos realizan conjuntamente la planificación y el seguimiento de trayectorias, es decir, reaccionan al entorno conforme navegan por éste. Por esta razón, no necesitan un algoritmo adicional para el seguimiento de trayectorias.

- **BUG**

El algoritmo BUG tiene como parámetro característico el sentido preestablecido para rodear obstáculos del entorno de trabajo cuando el robot se encuentra con ellos. En la figura 2.27 pueden observarse los menús de selección de este parámetro.

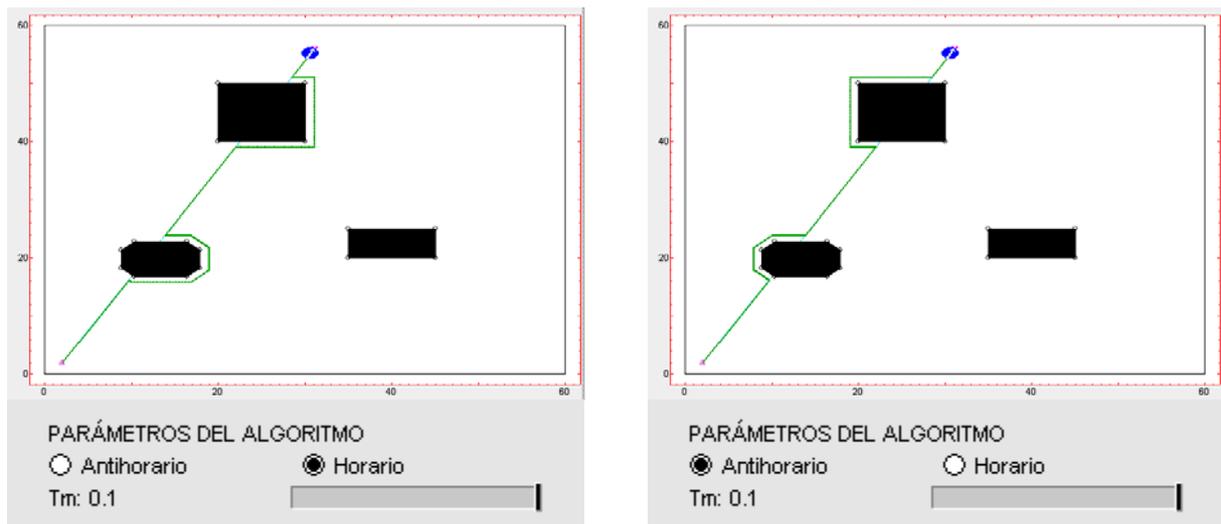


(a) Menú principal.

(b) Menú *Settings*.

Figura 2.27. Menús de selección del parámetro ‘sentido del giro’ en el algoritmo BUG.

En la figura 2.28 puede observarse el efecto del sentido de giro a la hora de rodear obstáculos.



(a) Sentido de giro horario.

(b) Sentido de giro antihorario.

Figura 2.28. Ejemplo del efecto de modificar el parámetro ‘sentido del giro’ en el algoritmo BUG.

- **VISBUG**

El algoritmo VISBUG tiene como parámetros característicos el sentido preestablecido para rodear obstáculos del entorno de trabajo cuando el robot se encuentra con ellos, y el alcance de los sensores de distancia del robot móvil (sónares). El robot muestra la orientación y alcance de los distintos sónares que incorpora dentro del entorno de trabajo. En la figura 2.29 puede observarse el cambio de apariencia del robot móvil cuando se modifica el alcance de sus sensores.

En la figura 2.30 pueden observarse los menús de selección del sentido de giro y el alcance de los sensores.

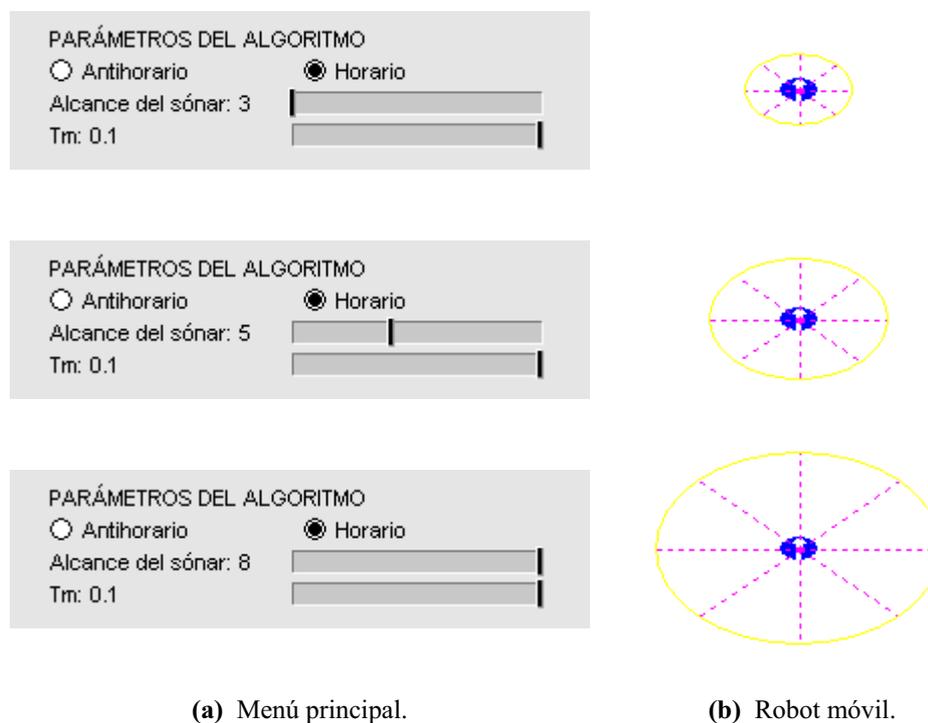
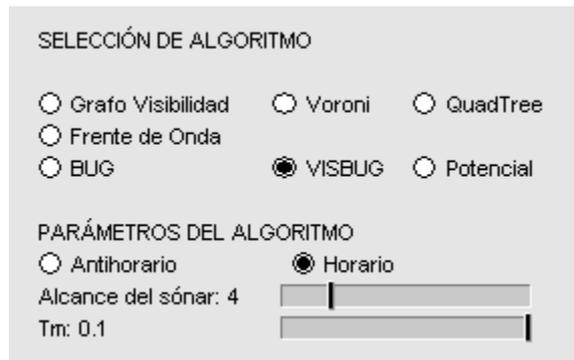
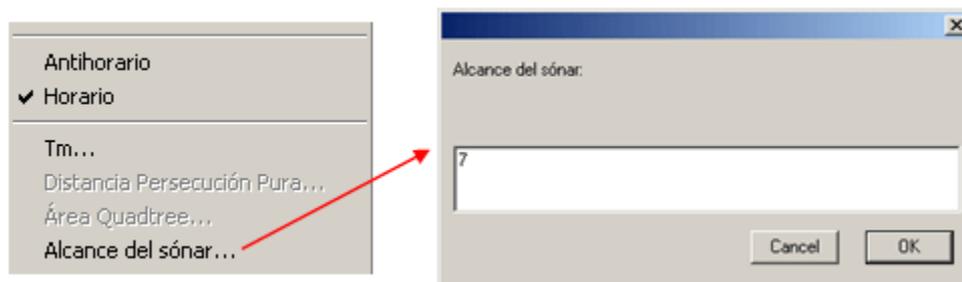


Figura 2.29. Ejemplos de varios valores para el parámetro 'alcance del s3nar'.

En la figura 2.31 puede observarse el efecto de modificar el alcance de los sensores del robot junto con el sentido de giro.

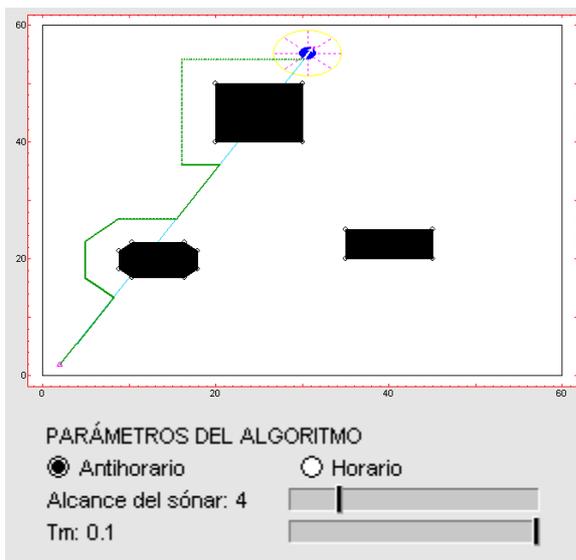


(a) Men3 principal.

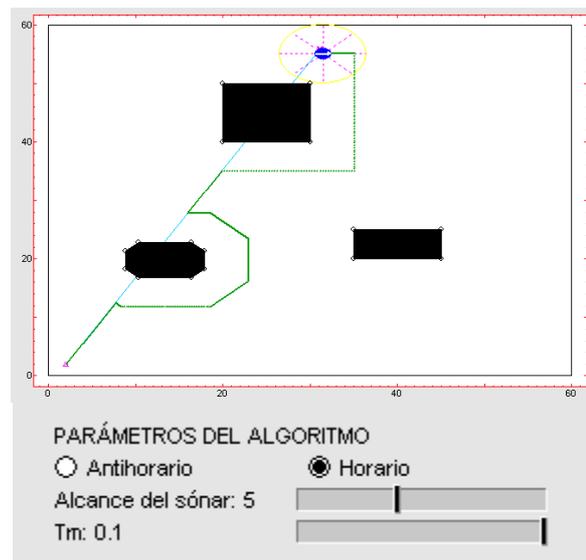


(b) Men3 Settings.

Figura 2.30. Men3 y ejemplo de selecci3n del par3metro ‘alcance del s3n3nar’.



(a) Sentido de giro antihorario, alcance del s3n3nar 4.

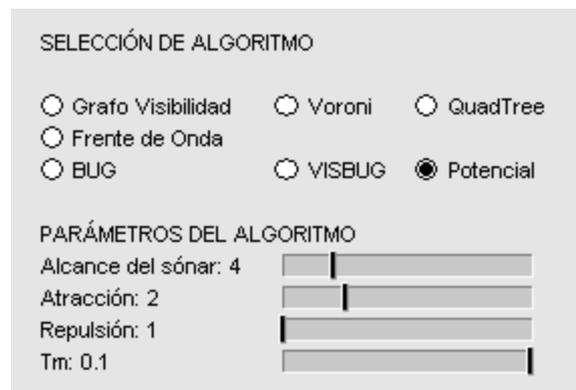


(b) Sentido de giro horario, alcance del s3n3nar 5.

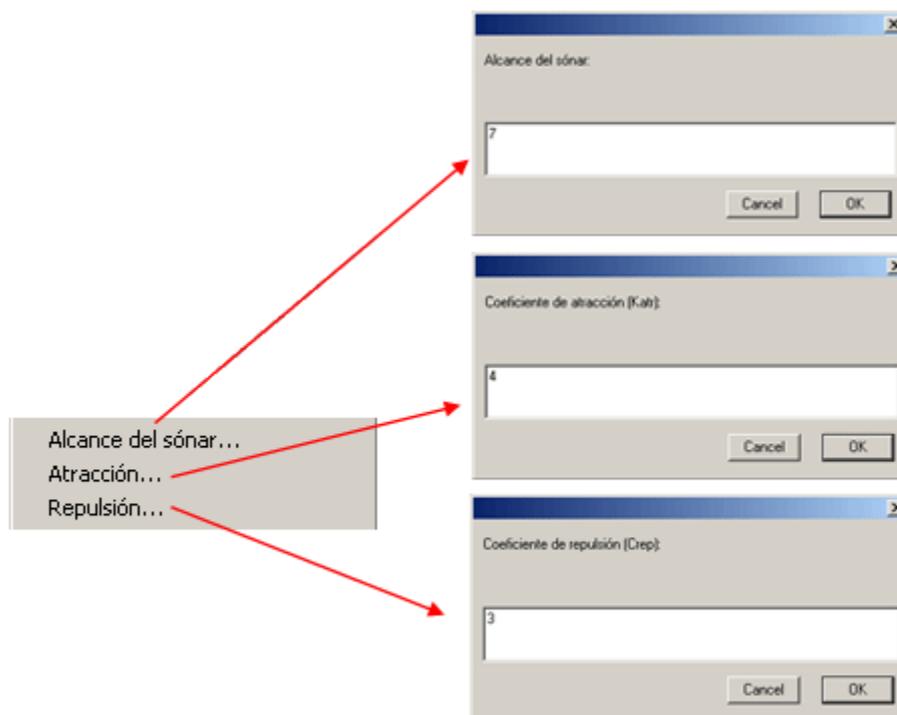
Figura 2.31. Ejemplo del efecto de modificar los par3metros ‘sentido de giro’ y ‘alcance del s3n3nar’.

- **Potencial**

El algoritmo Potencial tiene como parámetros característicos el alcance de los sensores de distancia del robot móvil (sónares), la constante de atracción al punto destino $Katr$, y la constante de repulsión a los obstáculos $Crep$. En la figura 2.32 pueden observarse los menús de selección de todos los parámetros característicos.



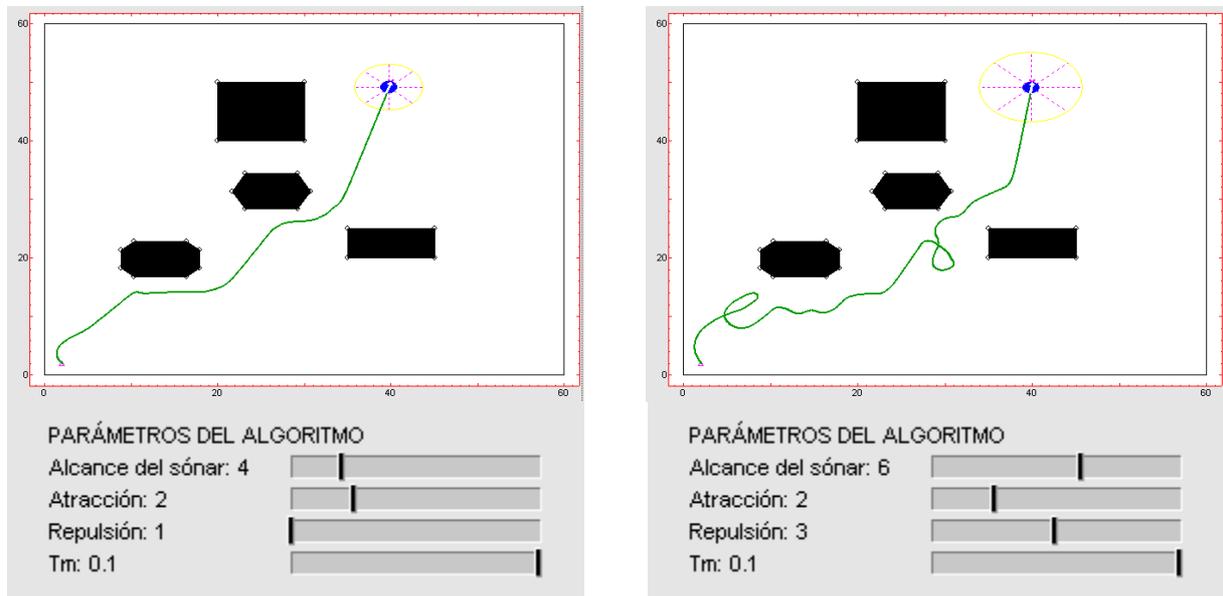
(a) Menú principal.



(b) Menú *Settings*.

Figura 2.32. Menús y ejemplo de selecci3n de los parámetros ‘alcance del s3nar’, ‘atracci3n’ y ‘repulsi3n’.

En la figura 2.33 puede observarse el efecto conjunto de modificar el alcance de los sensores, la constante de atracción, y la constante de repulsión.



(a) Alcance del s3nar 4, atracci3n 2 y repulsi3n 1.

(b) Alcance del s3nar 6, atracci3n 2 y repulsi3n 3.

Figura 2.33. Ejemplo de los efectos de modificar los par3metros ‘alcance del s3nar’, ‘atracci3n’ y ‘repulsi3n’.

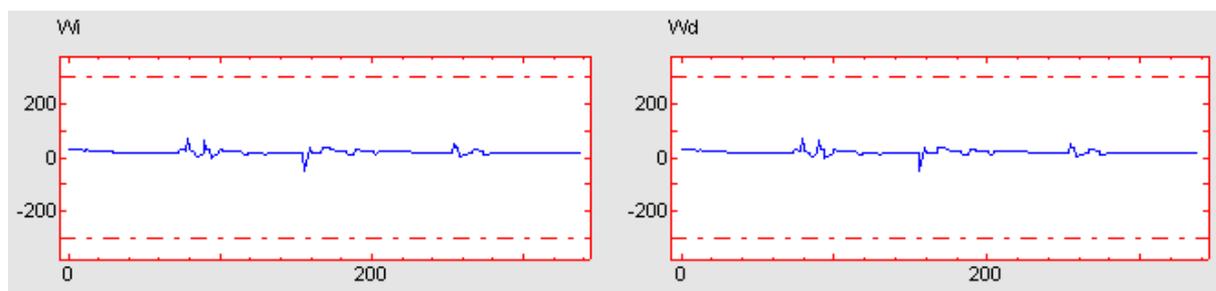
2.5. Cinem3ticas.

La herramienta dispone de tres cinem3ticas distintas: *Diferencial*, *Triciclo*, y *S3ncrona*. Las cinem3ticas tienen distintos par3metros caracter3sticos o se3ales de control: la tipo diferencial tiene las velocidades angulares de las ruedas del robot W_i y W_d , la tipo triciclo tiene el 3ngulo de la rueda libre, $Alpha$, y la tipo s3ncrono el 3ngulo de todas las ruedas, $Theta$.



(a) Men3 principal.

(b) Men3 Settings.



(c) Par3metros caracter3sticos de la cinem3tica diferencial ω_i y ω_d .

Figura 2.34. Men3s de selecci3n de la cinem3tica ‘diferencial’ y efecto sobre la interfaz de usuario.

Las gráficas correspondientes a los valores de las señales de control de cada cinemática se encuentran en la parte inferior derecha de la pantalla principal, justo debajo del entorno de trabajo. En las figuras 2.34, 2.35 y 2.36 se muestran los menús de selección de cada cinemática y las gráficas de sus parámetros característicos.

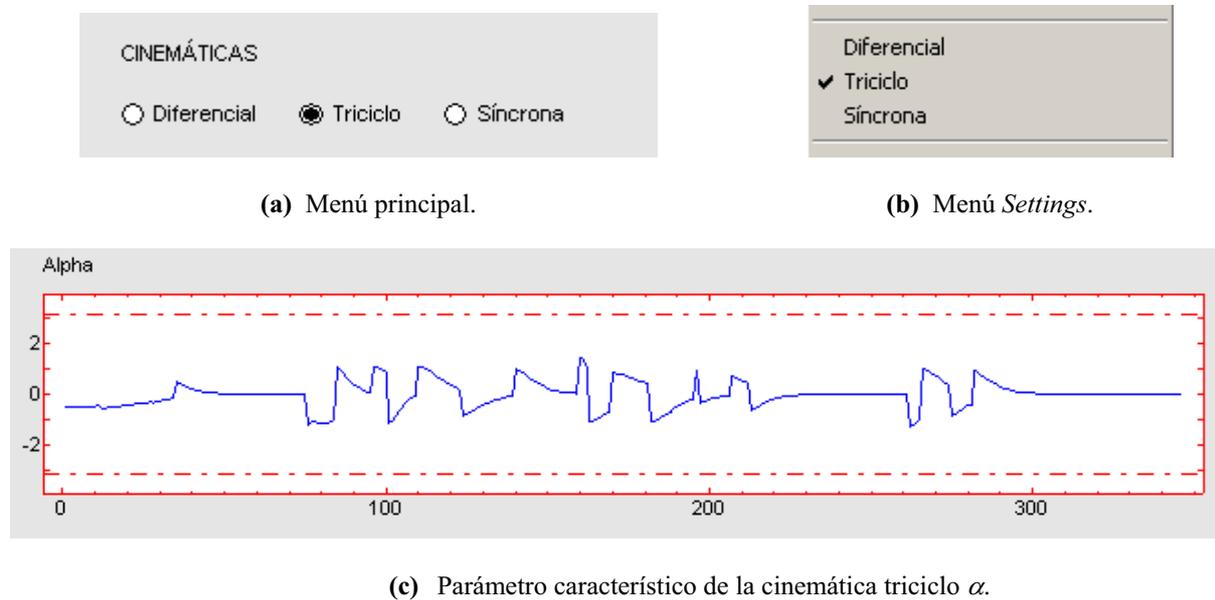


Figura 2.35. Menús de selección de la cinemática ‘triciclo’ y efecto sobre la interfaz de usuario.

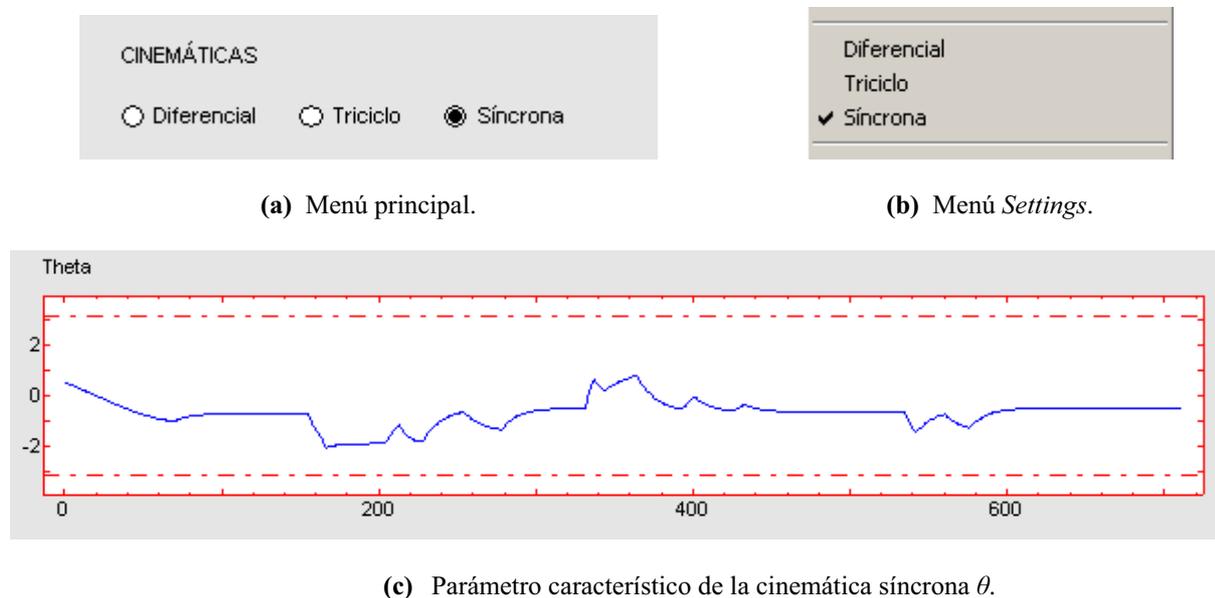
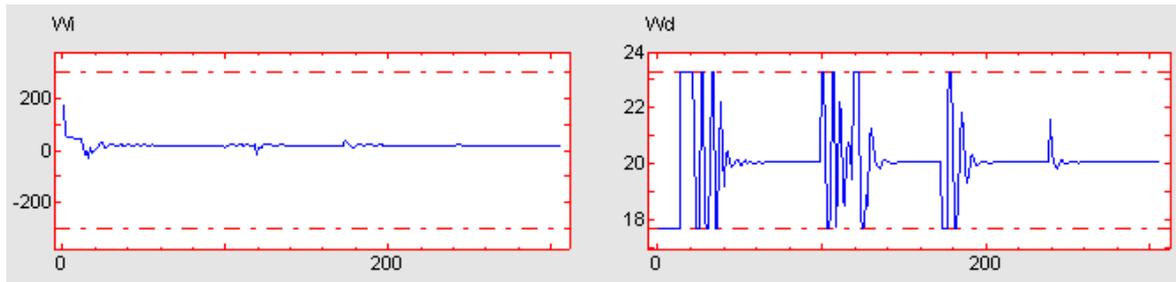
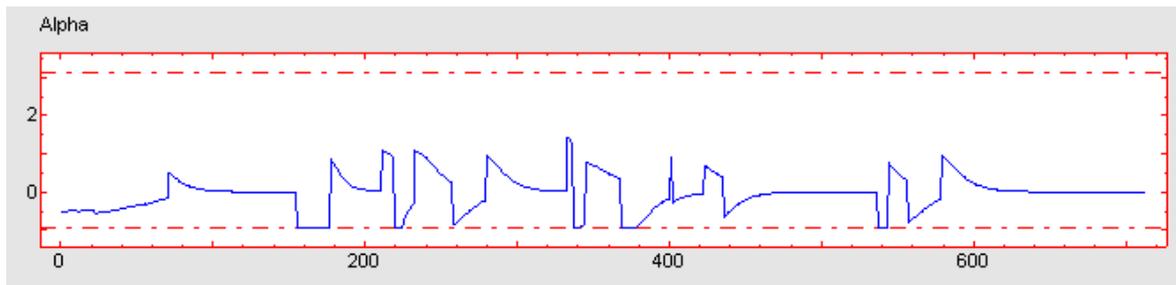


Figura 2.36. Menús de selección de la cinemática ‘síncrona’ y efecto sobre la interfaz de usuario.

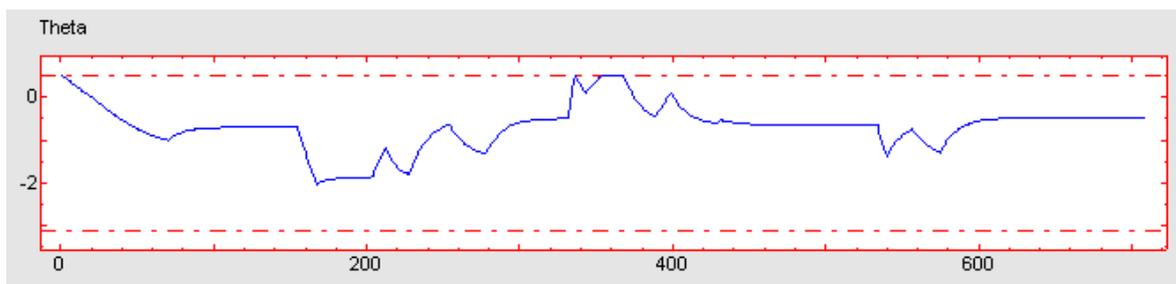
Las gráficas que representan los parámetros característicos de las cinemáticas cuentan con un límite inferior y un límite superior. El usuario puede situar estos límites donde desee, afectando así al seguimiento de la trayectoria del robot. El efecto de añadir restricciones en las distintas cinemáticas se muestra en la figura 2.37.



(a) Restricciones en cinemática diferencial.



(b) Restricciones en cinemática triciclo.



(c) Restricciones en cinemática síncrona.

Figura 2.37. Ejemplos del efecto de aplicar restricciones a las diferentes cinemáticas.

2.6. Simulación.

Para realizar la simulación en la herramienta interactiva debe seleccionarse el modo *Algoritmo*, y a continuación, mediante el menú *Settings*, seleccionar la simulación directa (*Ejecutar*) o *Paso a Paso*.

La simulación directa hace que el robot efectúe el seguimiento de la trayectoria planificada, mientras que la simulación paso a paso muestra los pasos seguidos durante la planificación. Así, en el modo *Paso a Paso*, para que el robot siga la trayectoria planificada durante los pasos, es necesario seleccionar la opción *Ejecutar*. La simulación *Paso a Paso* solo está disponible para los algoritmos globales, que son los únicos algoritmos que separan la planificación del seguimiento de trayectorias.

Para detener la simulación y volver al modo *Algoritmo*, tan solo es necesario volver a seleccionar el tipo de simulación que ya estaba seleccionada a través del menú *Settings*. En la figura 2.38 se muestran los menús de selección del tipo de simulación.

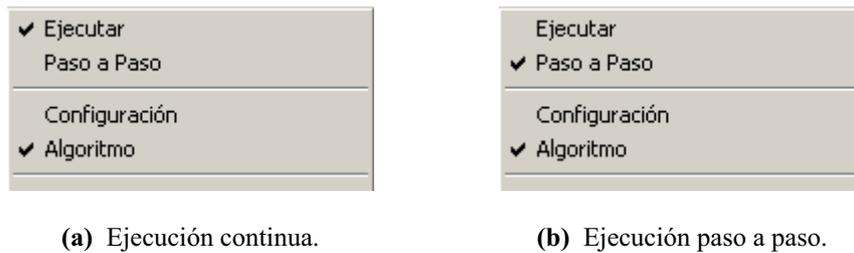
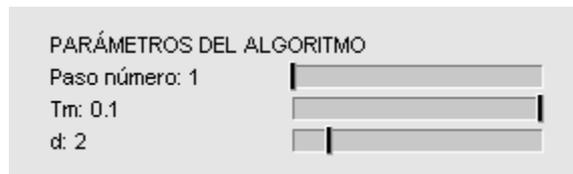


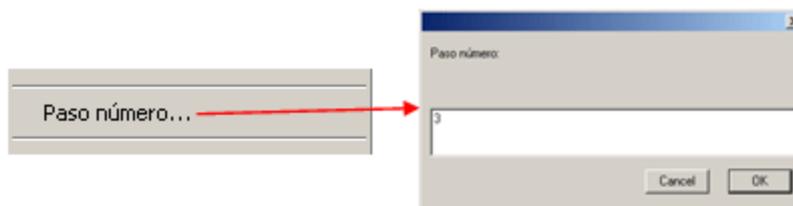
Figura 2.38. Menús de selección de los modos de ejecución de la simulación.

2.6.1 Simulación Paso a Paso.

Los pasos de simulación se muestran junto con los parámetros característicos de cada algoritmo, pudiendo seleccionar en cada momento el paso que se desee. En la figura 2.39 se muestran los menús de selección de los pasos de simulación.



(a) Menú principal.



(b) Menú *Settings*.

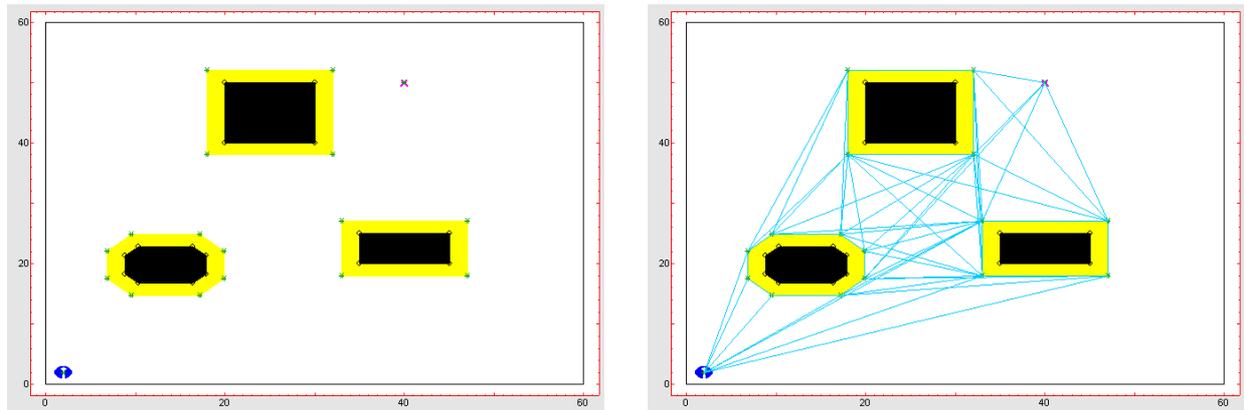
Figura 2.39. Menús y ejemplo de selección del parámetro 'paso actual'.

- **Grafo de Visibilidad.**

El algoritmo Grafo de Visibilidad cuenta con tres pasos:

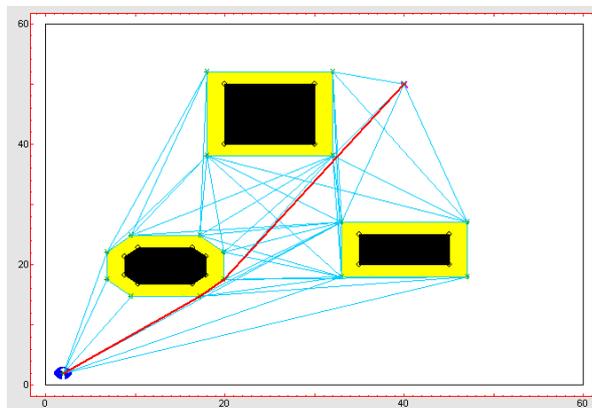
- 1) Puntos necesarios para calcular la matriz de visibilidad.
- 2) Todos los caminos posibles.
- 3) Camino mínimo.

En la figura 2.40 se muestran todos los pasos del algoritmo Grafo de Visibilidad.



(a) Paso 1. Vértices visibles.

(b) Paso 2. Todos los caminos posibles.



(c) Paso 3. Camino mínimo.

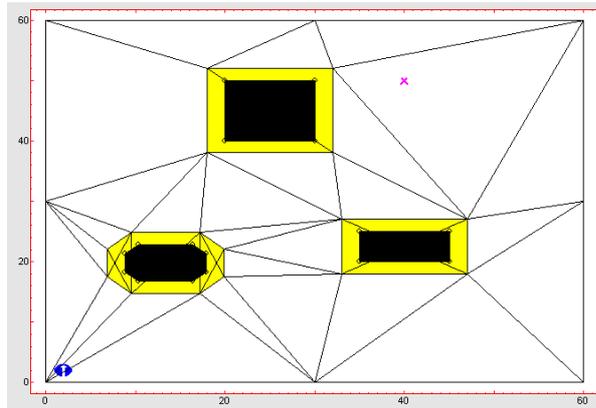
Figura 2.40. Pasos de simulación del algoritmo 'grafo de visibilidad'.

- **Diagrama de Voronoi.**

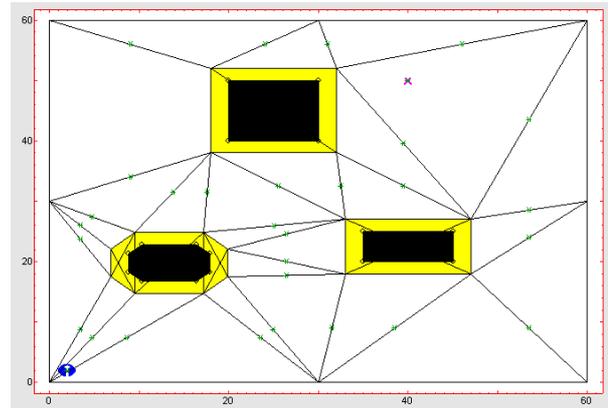
El algoritmo Diagrama de Voronoi cuenta con cuatro pasos:

- 1) Triangulaciones de Delaunay.
- 2) Puntos medios de las triangulaciones.
- 3) Todos los caminos posibles.
- 4) Camino mínimo.

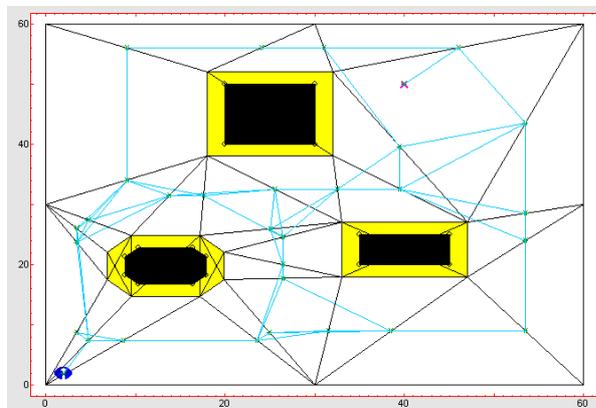
En la figura 2.41 se muestran todos los pasos del algoritmo Diagrama de Voronoi.



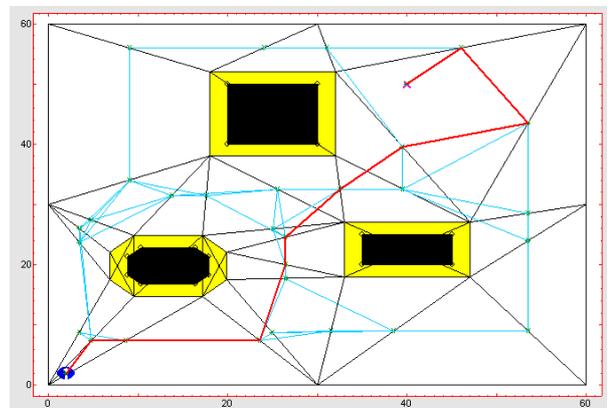
(a) Paso 1. Triangulaciones de Delaunay.



(b) Paso 2. Puntos medios de las triangulaciones.



(c) Paso 3. Todos los caminos posibles.



(d) Paso 4. Camino mínimo.

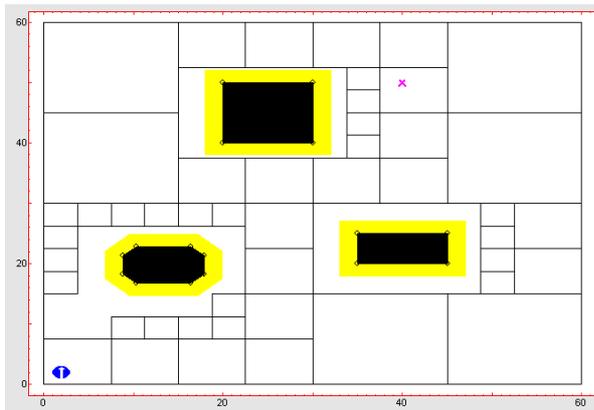
Figura 2.41. Pasos de simulación del algoritmo 'diagrama de Voronoi'.

- **Quadtree.**

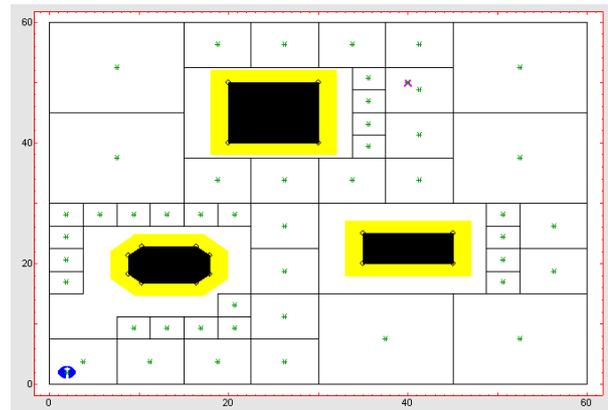
El algoritmo Quadtree cuenta con cuatro pasos:

- 1) División del entorno de trabajo en cuadrantes.
- 2) Puntos centrales de los cuadrantes.
- 3) Todos los caminos posibles.
- 4) Camino mínimo.

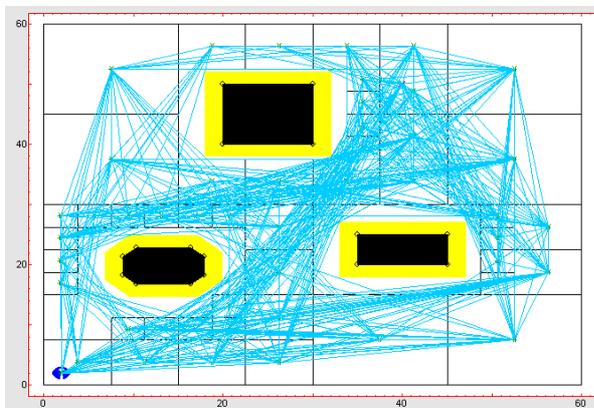
En la figura 2.42 se muestran todos los pasos del algoritmo Quadtree.



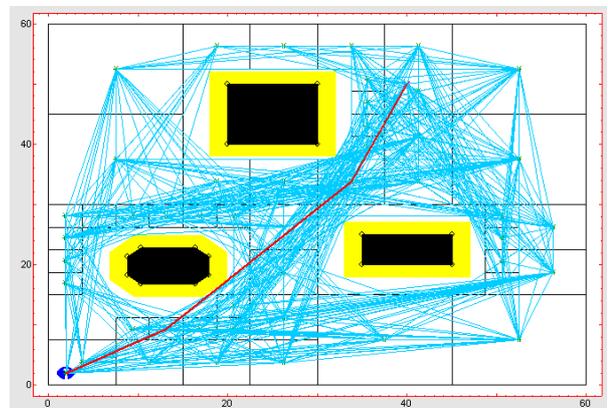
(a) Paso 1. Discretización del entorno de trabajo.



(b) Paso 2. Centroides de los cuadrantes.



(c) Paso 3. Todos los caminos posibles.



(d) Paso 4. Camino mínimo.

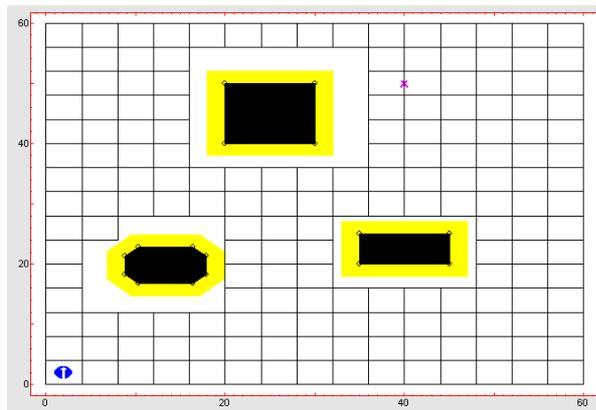
Figura 2.42. Pasos de simulación del algoritmo 'quad-tree'.

- **Frente de Onda.**

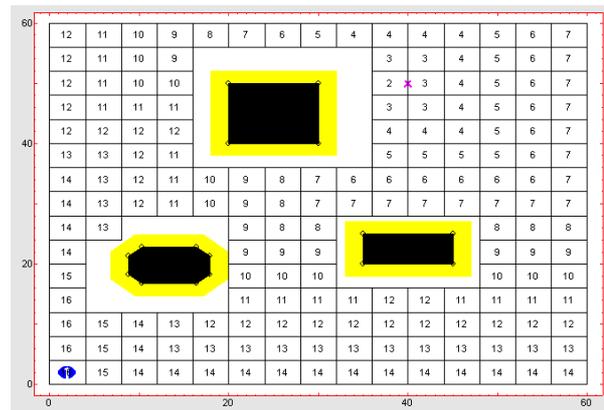
El algoritmo Frente de Onda cuenta con cuatro pasos:

- 1) División del entorno de trabajo en celdas.
- 2) Valores lógicos de las celdas.
- 3) Todos los caminos posibles.
- 4) Camino mínimo.

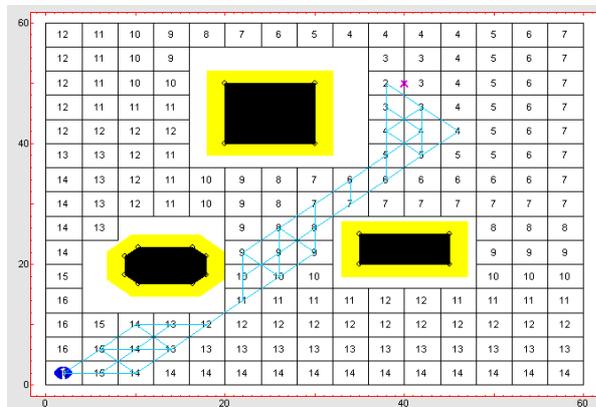
En la figura 2.43 se muestran todos los pasos del algoritmo Frente de Onda.



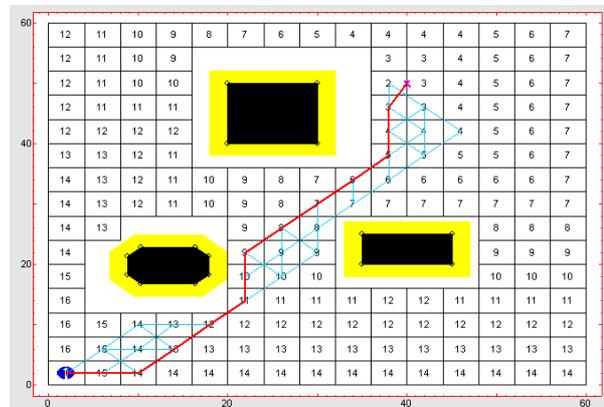
(a) Paso 1. Discretización del entorno de trabajo.



(b) Paso 2. Valores de las celdas.



(c) Paso 3. Todos los caminos posibles.



(d) Paso 4. Camino mínimo.

Figura 2.43. Pasos de simulación del algoritmo ‘frente de onda’.

2.6.2. Simulación en modo Ejecutar.

Este modo de funcionamiento será mostrado en la sección de ejemplos a través de varios supuestos prácticos.

2.6.3. Mensajes de Error.

A continuación se resumen los distintos mensajes de error que puede mostrar la herramienta y su significado:

- **Error al lanzar la ejecución.** Ocurre cuando se intenta lanzar una simulación desde el modo *Configuración* (figura 2.44).
- **Error de exceso de obstáculos.** Ocurre cuando hay 6 obstáculos en el entorno de trabajo y se intentan introducir más (figura 2.45).



Figura 2.44. Mensaje de error al lanzar la ejecución de la simulación.



Figura 2.45. Mensaje de error de exceso de obstáculos en el entorno de trabajo.

- **Error de colisión.** Ocurre cuando el robot móvil choca contra un obstáculo (figura 2.46).



Figura 2.46. Mensaje de error de colisión del robot móvil.

- **Error de mínimo local.** Ocurre cuando el robot móvil con algoritmo Potencial no consigue superar un obstáculo (figura 2.47).



Figura 2.47. Mensaje de error de mínimo local en el entorno de trabajo.

3. Ejemplos.

En este apartado se presentan a modo de ejemplo algunos de los resultados obtenidos con la herramienta de simulación interactiva MRIT. Se muestran ejemplos de simulación de los distintos algoritmos de planificación y seguimiento de trayectorias, así como de las distintas cinemáticas, de forma individual y de forma comparativa.

3.1. Análisis y comparación de algoritmos de navegación.

Los distintos algoritmos presentes en la herramienta ofrecen diversos resultados ante un mismo problema de planificación. A lo largo de esta sección se presentan varios ejemplos comparativos donde se utilizan algoritmos globales y locales, mostrando que es posible estudiar cual de ellos presenta mejores características frente al problema de navegación dentro de un entorno de trabajo específico.

A continuación, se muestran ejemplos de cada uno de los algoritmos de navegación y seguimiento de trayectorias, en los que se puede observar el efecto de modificar los valores de los parámetros característicos de cada algoritmo.

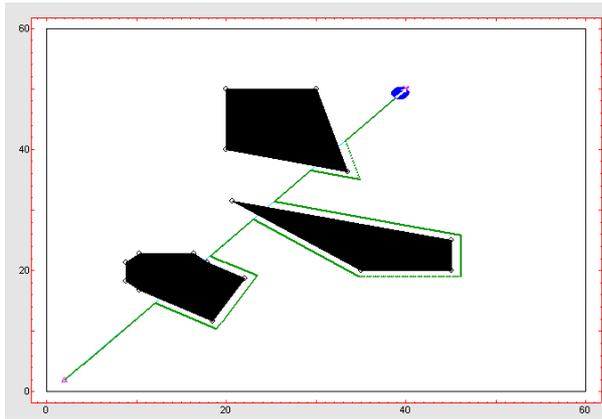
3.1.1 Algoritmos locales.

3.1.1.1. BUG.

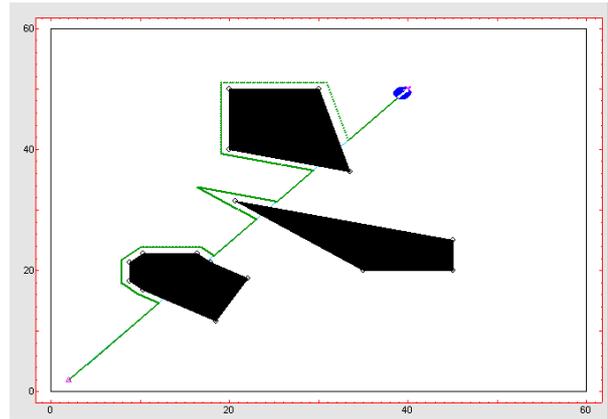
Como se describió anteriormente, el algoritmo *BUG* tiene como parámetro característico el sentido de giro a la hora de rodear obstáculos. Con el fin de observar la influencia de este parámetro, en las figuras 3.1 a 3.3 se muestran diversos ejemplos para ambos sentido de giro, afectados también por modificaciones en el cambio de tamaño del robot móvil.

3.1.1.2. VISBUG.

El lo que se refiere al algoritmo *VISBUG*, existen dos parámetros característicos, el sentido del giro a la hora de rodear los obstáculos, y el alcance de los sónares (sensores de distancia). Ejemplos de las modificaciones de de estos parámetros son mostrados en las figuras 3.4 a 3.6. En estas figuras se puede observar el efecto que tiene el cambio de tamaño del robot, junto con el radio de alcance de los sónares y el sentido de giro (se modifican conjuntamente el alcance del sónar y el tamaño del robot porque, al aumentar este último, aumenta el mínimo valor del alcance de sónar). Cabe destacar que, en casos como el de la figura 3.6 (b), si el objeto a evitar y el alcance de los sónares es demasiado grande, el robot móvil puede colisionar con los límites del entorno de trabajo.

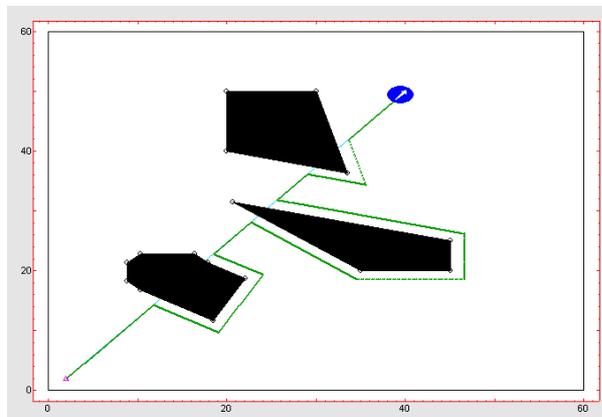


(a) Sentido de giro antihorario.

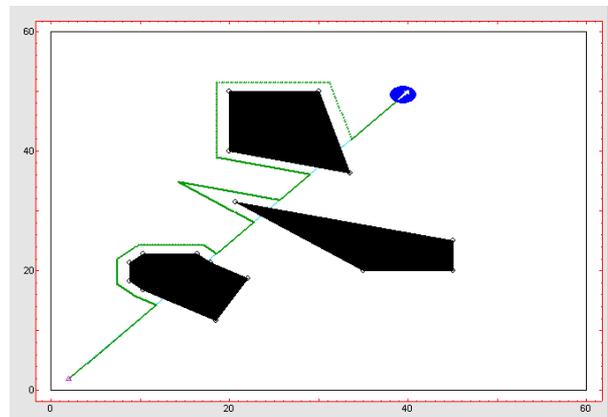


(b) Sentido de giro horario.

Figura 3.1. Algoritmo BUG con tamaño del robot 2.

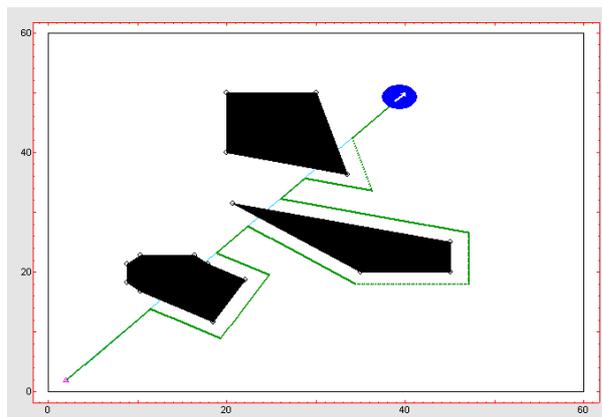


(a) Sentido de giro antihorario.

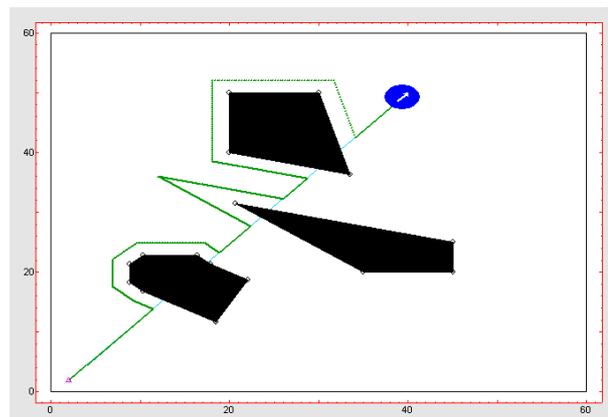


(b) Sentido de giro horario.

Figura 3.2. Algoritmo BUG con tamaño del robot 3.

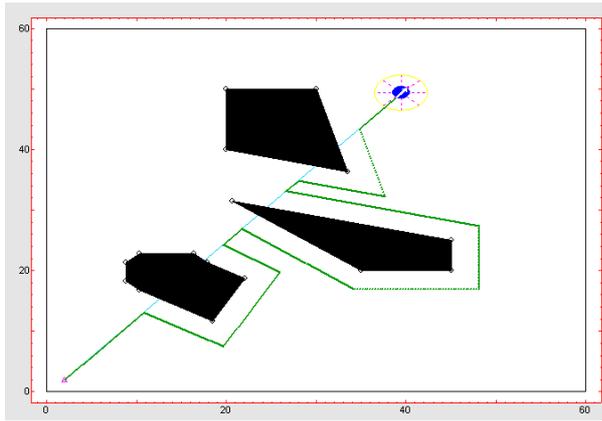


(a) Sentido de giro antihorario.

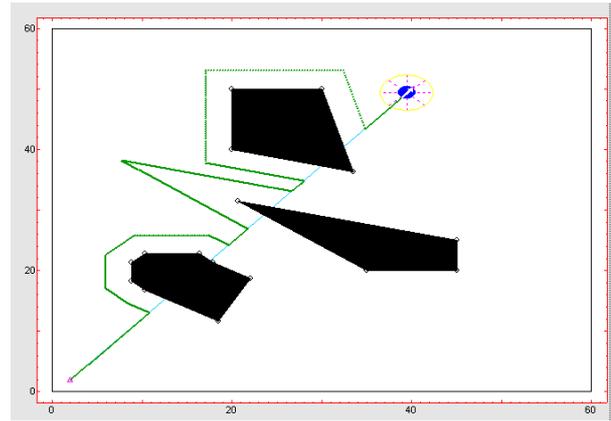


(b) Sentido de giro horario.

Figura 3.3. Algoritmo BUG con tamaño del robot 4.

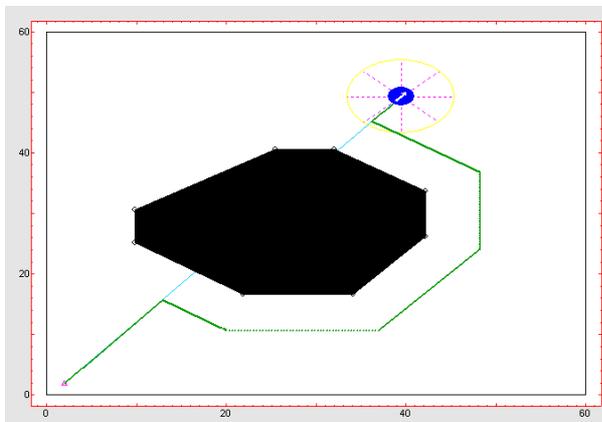


(a) Sentido de giro antihorario.

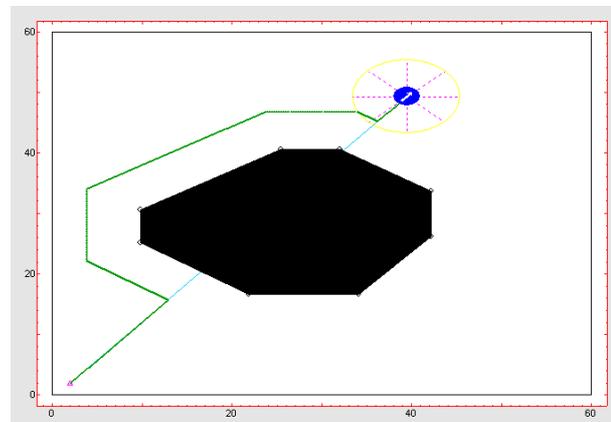


(b) Sentido de giro horario.

Figura 3.4. Algoritmo VISBUG con alcance del s3n3n3r 3 y tama1o del robot 2.

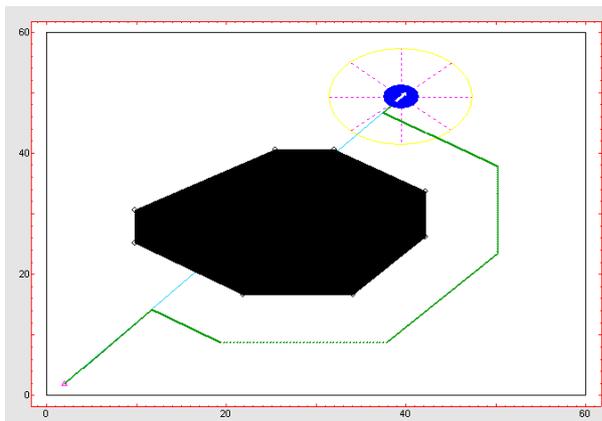


(a) Sentido de giro antihorario.

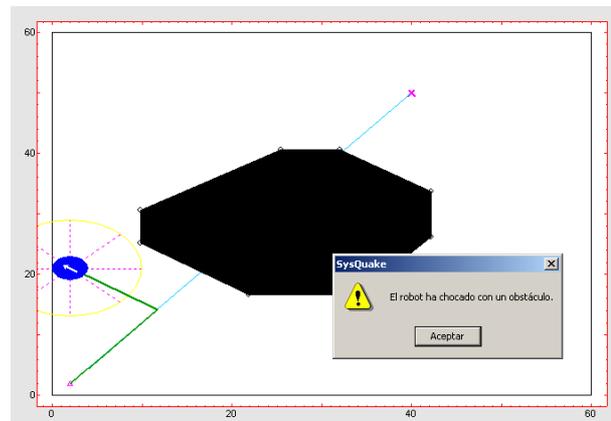


(b) Sentido de giro horario.

Figura 3.5. Algoritmo VISBUG con alcance del s3n3n3r 6 y tama1o del robot 3.



(a) Sentido de giro antihorario.



(b) Sentido de giro horario.

Figura 3.6. Algoritmo VISBUG con alcance del s3n3n3r 8 y tama1o del robot 4.

3.1.1.3. Potencial.

El algoritmo *Potencial* posee como parámetros característicos el alcance de los sónares, la constante de atracción, y la constante de repulsión. En la figura 3.7 se puede observar el efecto de modificar las constantes de atracción y repulsión. Como se puede ver, el valor de las constantes no puede tener una diferencia grande, ya que puede provocar que el robot móvil colisione, al primar el avance hacia el objetivo respecto de la evitación de los obstáculos.

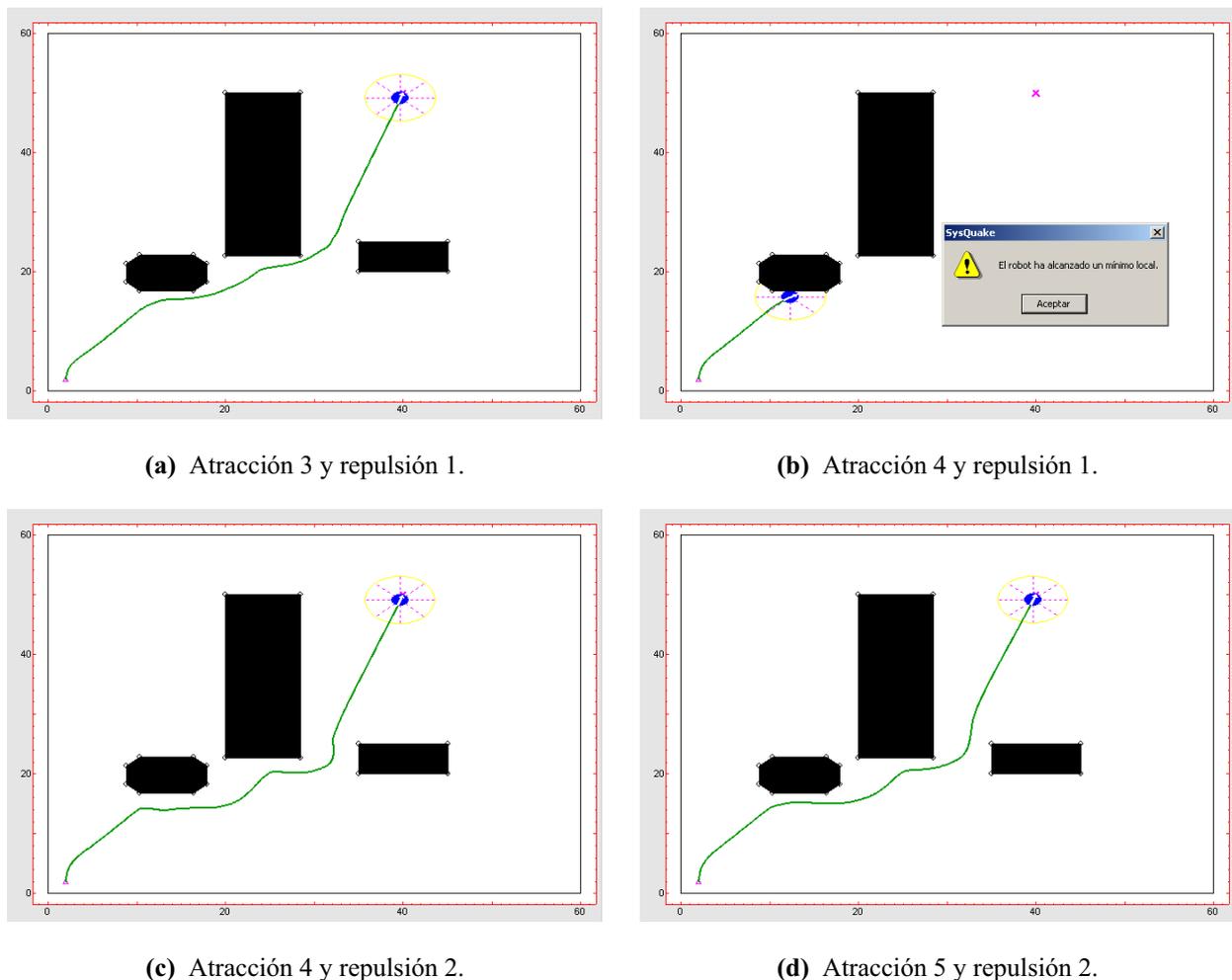
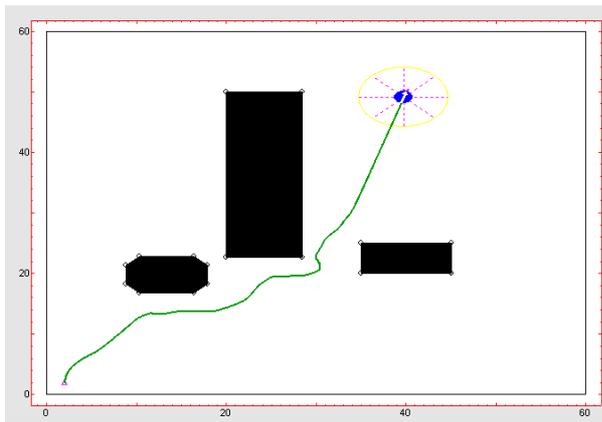


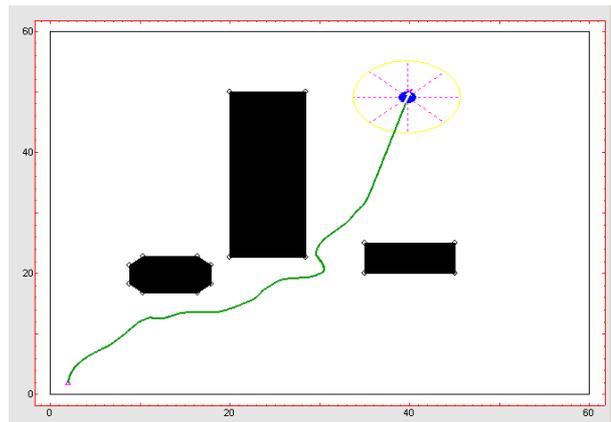
Figura 3.7. Algoritmo Potencial con alcance del s3nar 4.

En la figura 3.8 puede observarse el efecto de variar el alcance de los s3nares del robot m3vil. Cabe destacar como, a mayor alcance de sensores de distancia, el robot m3vil reacciona mejor ante la presencia de obst3culos, consiguiendo una trayectoria m3s alejada de 3stos y, por tanto, m3s segura.

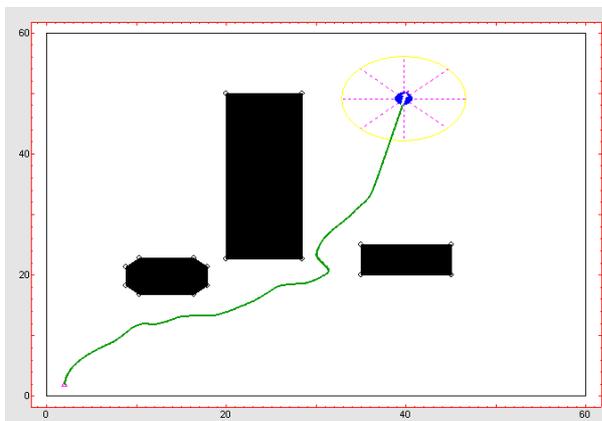
Con el fin de mostrar posibles problemas, en la figura 3.9 se observa un ejemplo donde, a pesar de tener un alcance bastante alto de s3nares, el robot m3vil es demasiado grande y colisiona al intentar pasar entre dos obst3culos cercanos entre si.



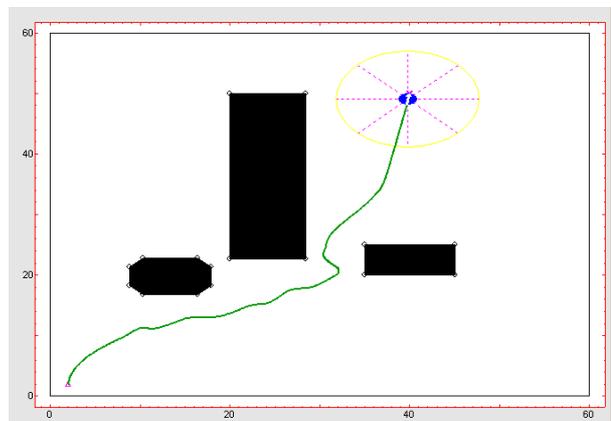
(a) Alcance del s3n3r 5.



(b) Alcance del s3n3r 6.



(c) Alcance del s3n3r 7.



(d) Alcance del s3n3r 8.

Figura 3.8. Algoritmo Potencial con atracci3n 2 y repulsi3n 1.

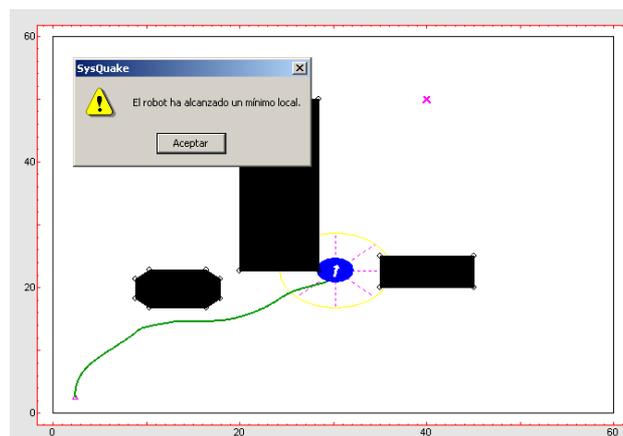


Figura 3.9. Algoritmo Potencial con alcance del s3n3r 6, atracci3n 5, repulsi3n 2 y tama3o 4.

3.1.2. Algoritmos globales.

3.1.2.1. Grafo de visibilidad.

El algoritmo *Grafo de Visibilidad* no tiene ningún parámetro característico, por tanto, con miras a mostrar su funcionamiento en la figura 3.10 se muestra el efecto que provoca la modificación del tamaño del robot móvil sobre los resultados del algoritmo. De éstos ejemplos se puede observar que cuanto más ancho es el robot móvil, más se engrosan los obstáculos y, por lo tanto, más curva es la trayectoria calculada.

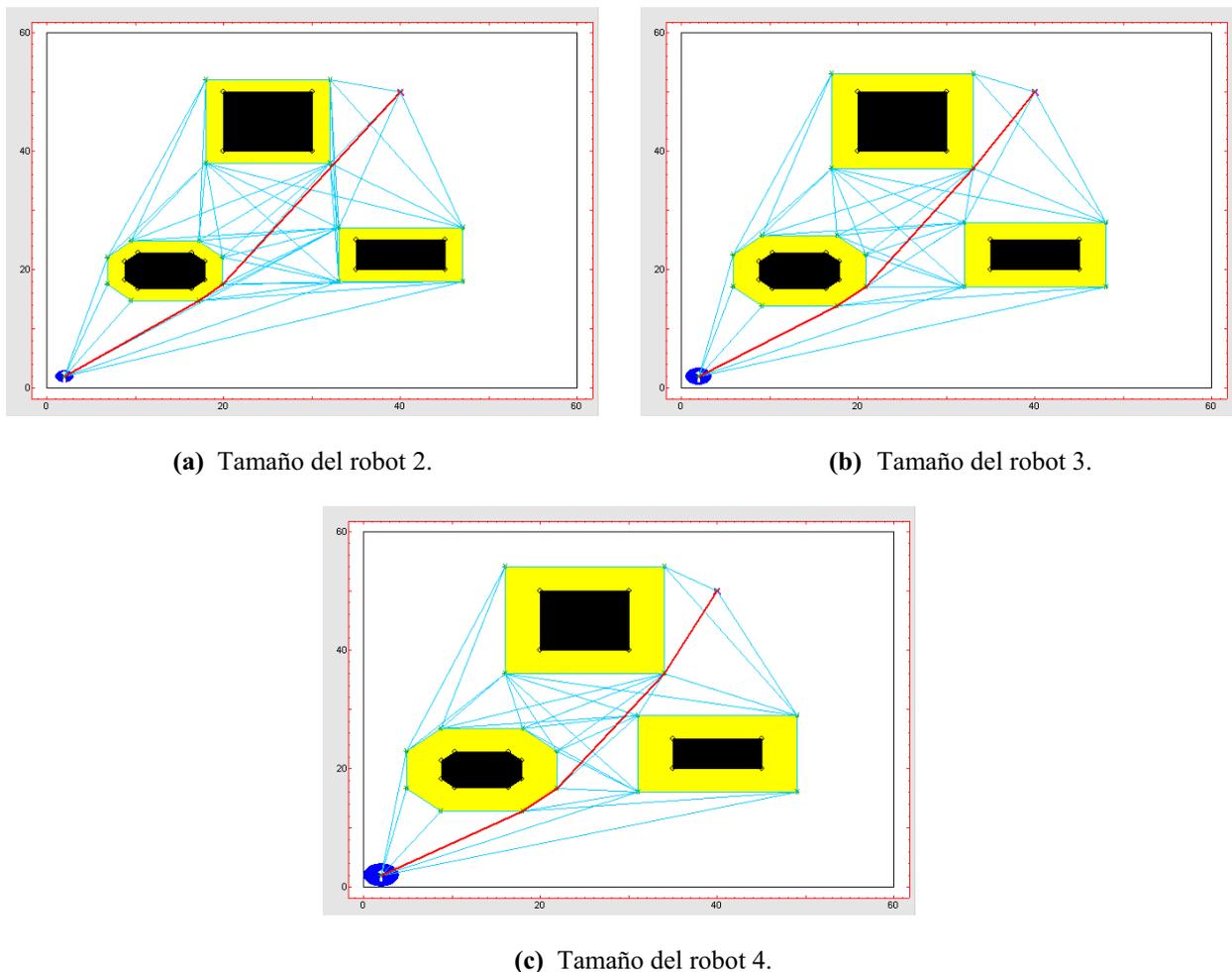
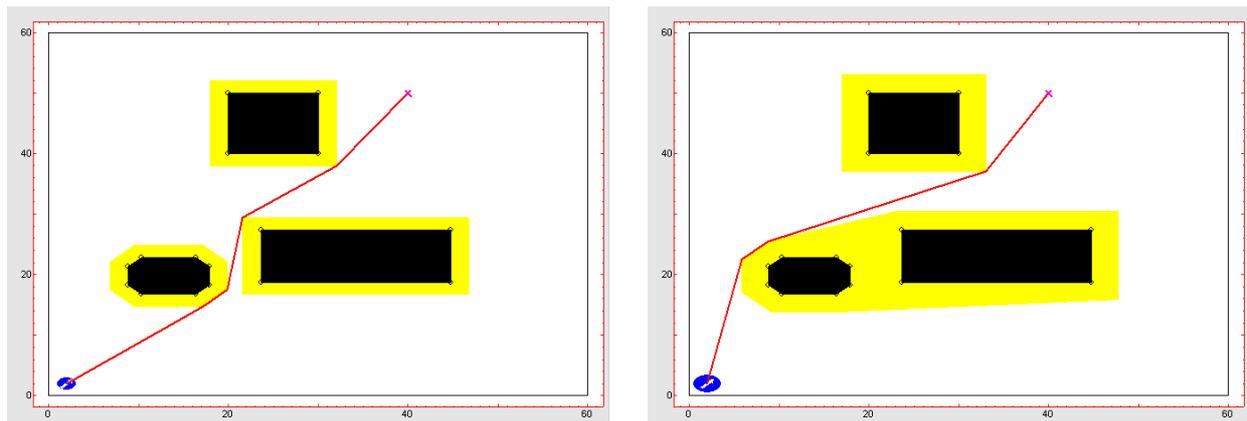


Figura 3.10. Algoritmo grafo de visibilidad con distintos tamaños de robot.

El efecto de aumentar el tamaño del robot móvil, junto con el engrosamiento correspondiente de los obstáculos, puede dar lugar a la fusión de obstáculos como puede observarse en el ejemplo de la figura 3.11. Nótese como la cercanía de los obstáculos de la parte inferior del entorno de trabajo hace que se fusionen cuando se aumenta el tamaño del robot, dando lugar a trayectorias diferentes.



(a) Tamaño del robot 2.

(b) Tamaño del robot 3.

Figura 3.11. Fusión de obstáculos con algoritmo grafo de visibilidad.

3.1.2.2. Diagrama de Voronoi.

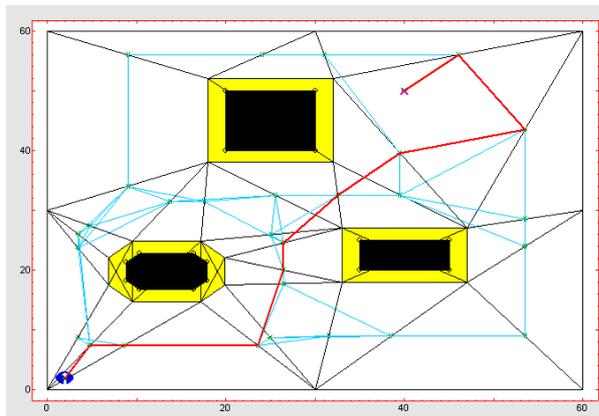
Al igual que el algoritmo *Grafo de Visibilidad*, el algoritmo basado en el *Diagrama de Voronoi* tampoco posee ningún parámetro característico. En la figura 3.12 puede observarse el efecto del crecimiento del tamaño del robot. Lo que se observa en los ejemplos es cómo el algoritmo hace que, en cada caso, la trayectoria se ajuste al ancho de los obstáculos y del robot móvil, consiguiendo que se mantenga lo más alejada posible de los obstáculos y de los límites del entorno de trabajo.

Cabe destacar como la trayectoria calculada en la figura 3.12 (b) es totalmente diferente a las otras dos. Esto se debe a que, al aumentar el tamaño de los obstáculos, varía el número de posibles caminos y, por lo tanto, el camino mínimo puede ser totalmente distinto al anterior a dicha modificación.

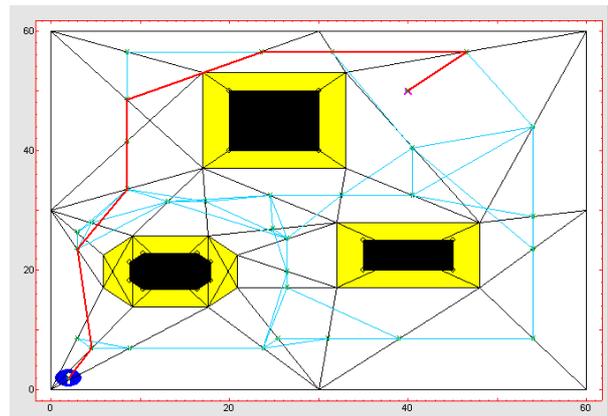
3.1.2.3. Quadtree.

El algoritmo *Quadtree* tiene como parámetro característico el tamaño del área de la cuadrícula. En las figuras 3.13 a 3.15 se observa el efecto de aumentar el tamaño de cuadrícula junto con el tamaño del robot.

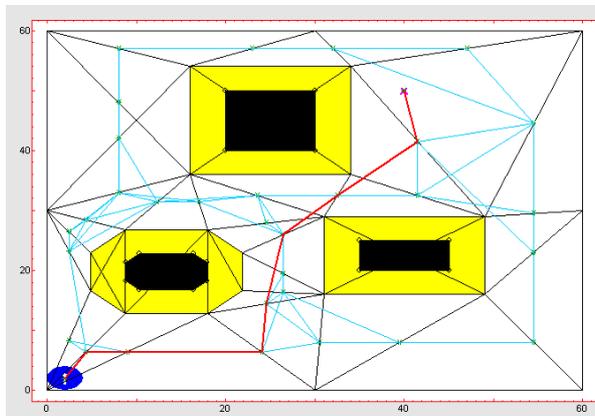
Puede deducirse que, a mayor tamaño de cuadrícula, mayor eficiencia en el cálculo de la trayectoria, siempre que el entorno de trabajo no contenga demasiados obstáculos. De la misma forma, se puede observar cómo esa ineficiencia al aumentar el tamaño de la cuadrícula se ve reflejada en caminos resultantes con cambios bruscos en la trayectoria.



(a) Tamaño del robot 2.

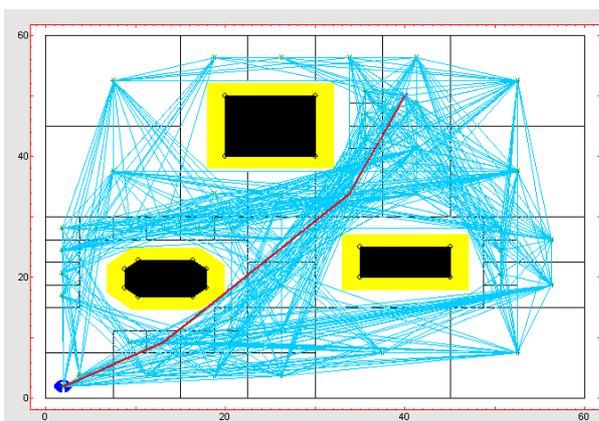


(b) Tamaño del robot 3.

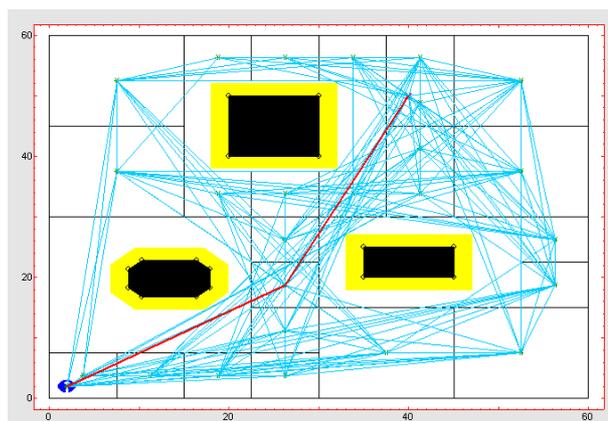


(c) Tamaño del robot 4.

Figura 3.12. Algoritmo Diagrama de Voronoi con distintos tamaños de robot.



(a) Área quadtree 1.



(b) Área quadtree 2.

Figura 3.13. Algoritmo quadtree con distintas áreas quadtree y tamaño de robot 2.

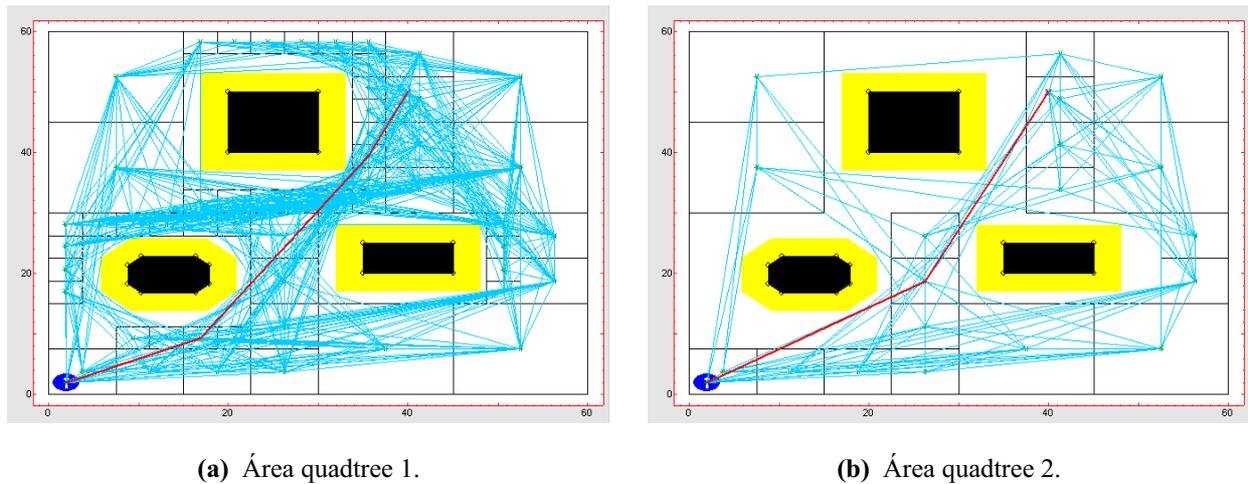


Figura 3.14. Algoritmo quadtree con distintas áreas quadtree y tamaño de robot 3.

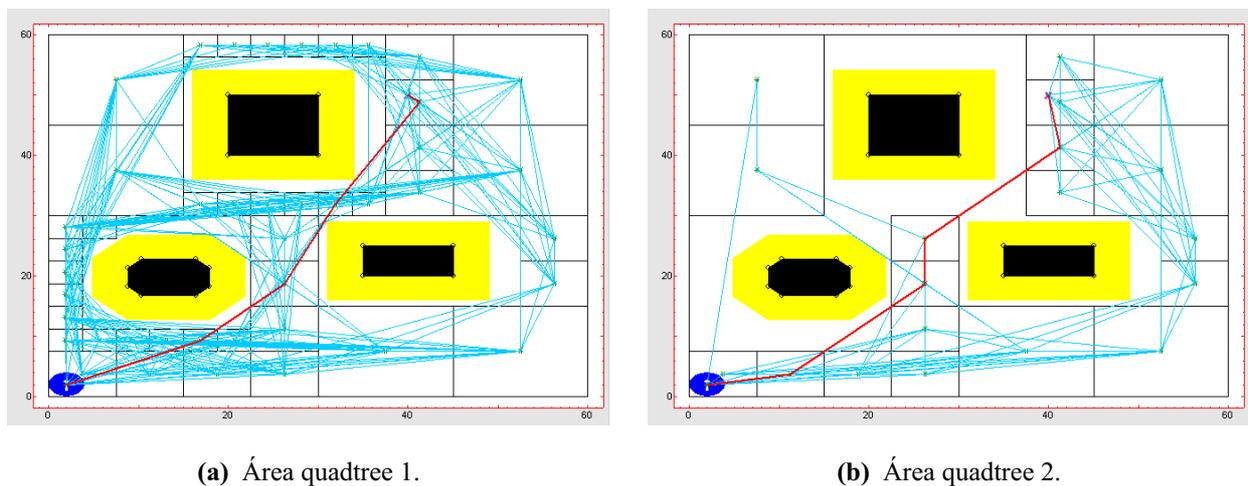
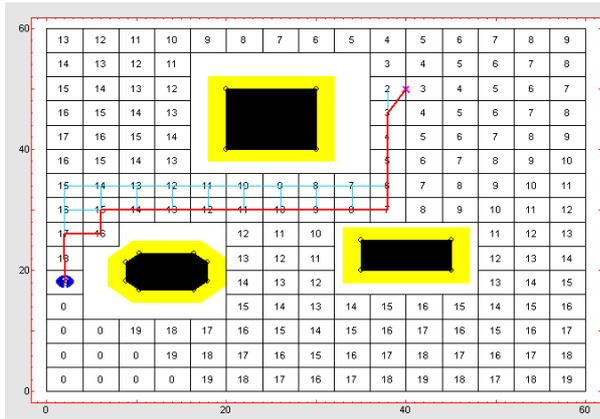


Figura 3.15. Algoritmo quadtree con distintas áreas quadtree y tamaño de robot 4.

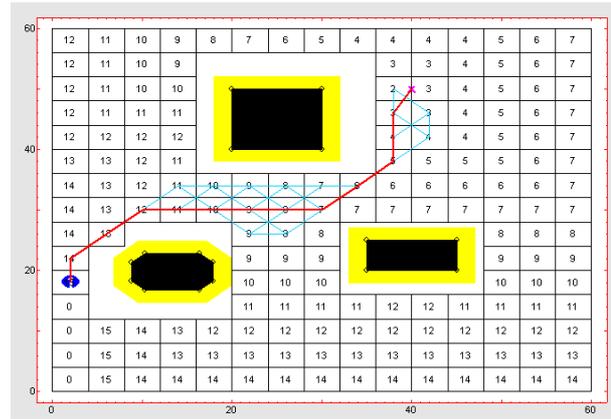
3.1.2.4. Frente de onda.

El último de los algoritmos globales es el *Frente de Onda*, el cual posee como parámetros característicos el tamaño de celda y la vecindad entre éstas. En las figuras 3.16 a 3.18 se observan distintos ejemplos en los que se modifican conjuntamente el tamaño de las celdas y su vecindad.

Como conclusión cabe destacar que cuando se utiliza vecindad 8, la trayectoria obtenida es mucho más suave que con vecindad 4, ya que ésta última no considera los posibles segmentos diagonales del camino.

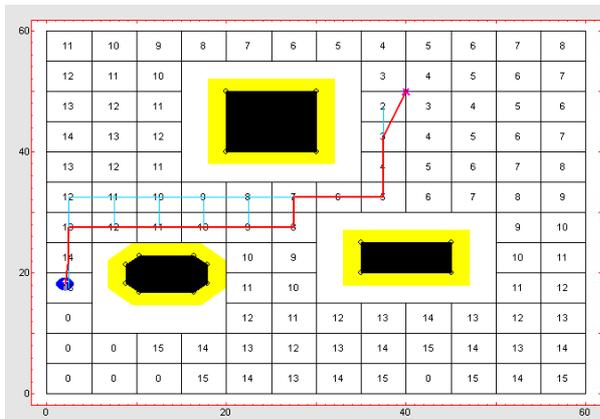


(a) Vecindad 4.

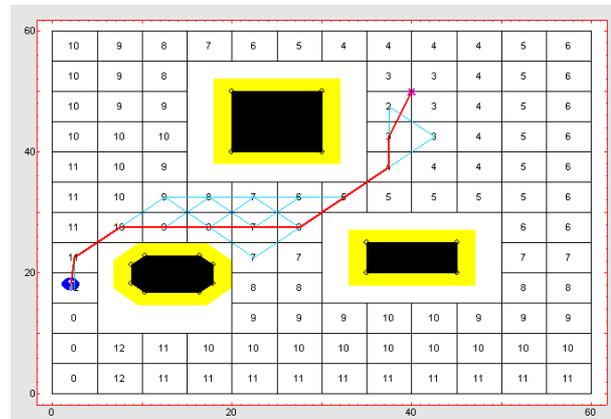


(b) Vecindad 8.

Figura 3.16. Algoritmo frente de onda con tamaño de celda 4 y tamaño de robot 2.

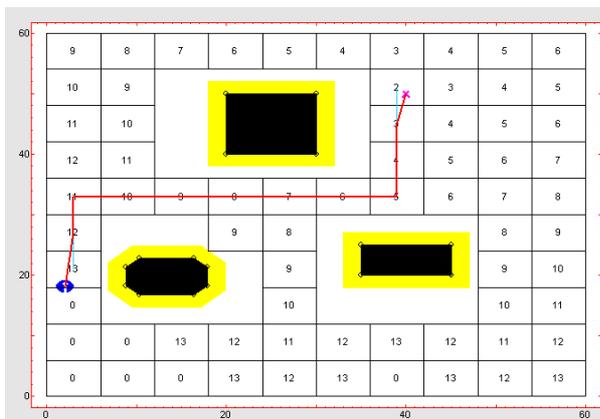


(a) Vecindad 4.

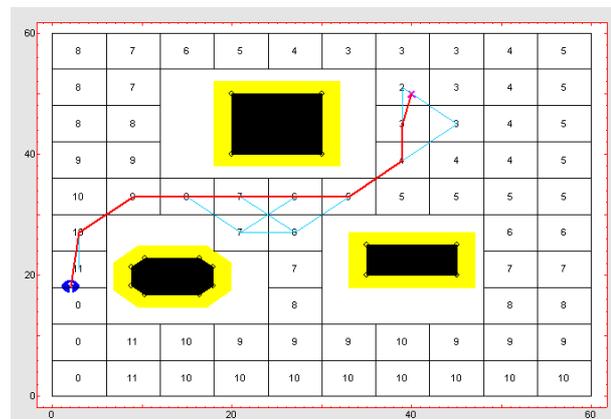


(b) Vecindad 8.

Figura 3.17. Algoritmo frente de onda con tamaño de celda 5 y tamaño de robot 2.



(a) Vecindad 4.



(b) Vecindad 8.

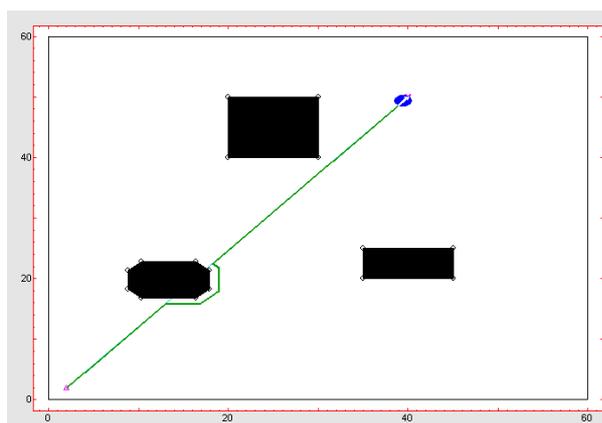
Figura 3.18. Algoritmo frente de onda con tamaño de celda 6 y tamaño de robot 2.

3.1.3. Comparación de algoritmos de navegación.

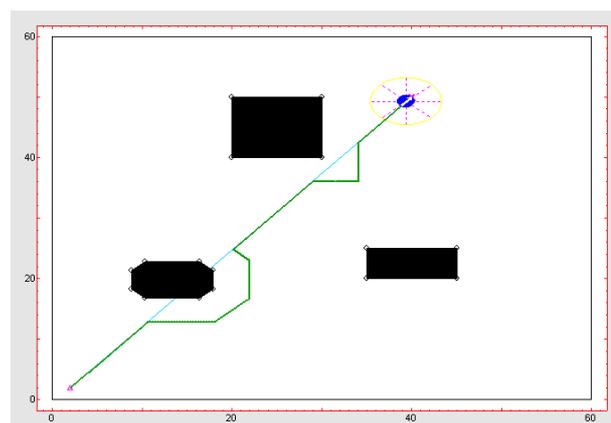
Esta sección muestra los distintos resultados que pueden ser obtenidos con los distintos algoritmos de navegación para un mismo entorno. Se muestra un determinado entorno y se comparan los resultados obtenidos para los algoritmos locales y globales.

A la hora de comparar los resultados obtenidos y decidir cual de los distintos algoritmos es el mejor apropiado, es necesario establecer los criterios de evaluación de resultados. En el campo de la robótica móvil los criterios más utilizados se suelen basar en la longitud del camino, la suavidad en la curvatura y la lejanía a los obstáculos (más seguridad). De esta manera, los resultados mostrados en los ejemplos se comparan heurísticamente teniendo en cuenta estas características, ya que no se ha llevado a cabo el cálculo exacto de ellas (salvo en el caso de la distancia para los algoritmos de planificación global, donde se calcula siempre la trayectoria de menor longitud).

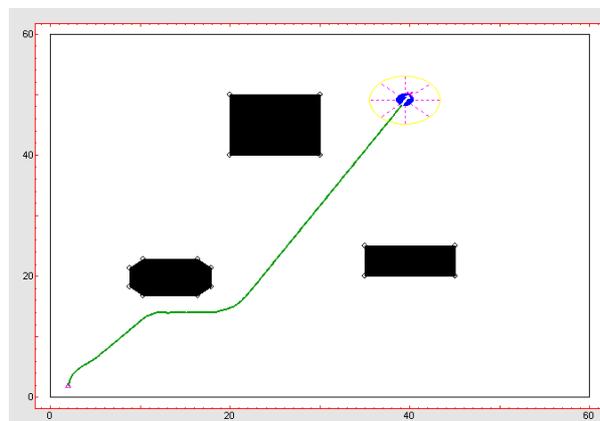
En primer lugar se muestran los resultados obtenidos haciendo uso de los distintos algoritmos reactivos. Tales resultados se muestran en la figura 3.19.



(a) Algoritmo BUG.



(b) Algoritmo VISBUG.

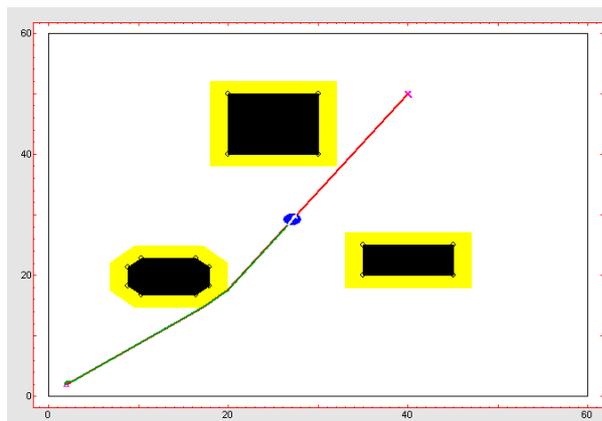


(c) Algoritmo Potencial.

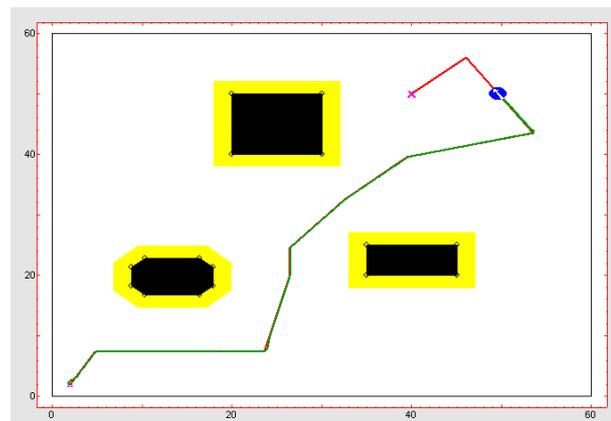
Figura 3.19. Ejemplo de los algoritmos de planificación local.

Teniendo en cuenta los criterios comentados anteriormente, se observa que el algoritmo BUG es el de menor longitud, el Potencial el más suave y el VISBUG el más alejado de los obstáculos del entorno. A la hora de seleccionar el algoritmo más apropiado habrá que tener en cuenta el entorno de trabajo.

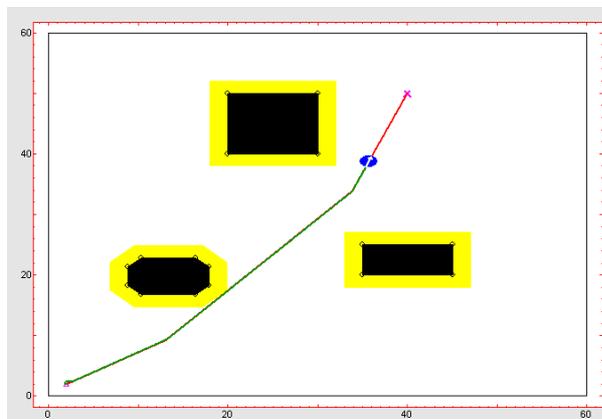
En la figura 3.20 se observa un ejemplo comparativo de los algoritmos de planificación global dentro del mismo entorno de trabajo. Como se puede observar los caminos mínimos encontrados por estos algoritmos dejan constancia del conocimiento previo del entorno por parte de los mismos. El Grafo de visibilidad y Quadtree ofrecen los caminos más cortos y de curvatura más suave, mientras que el Diagrama de Voronoi consigue el camino más alejado de los obstáculos.



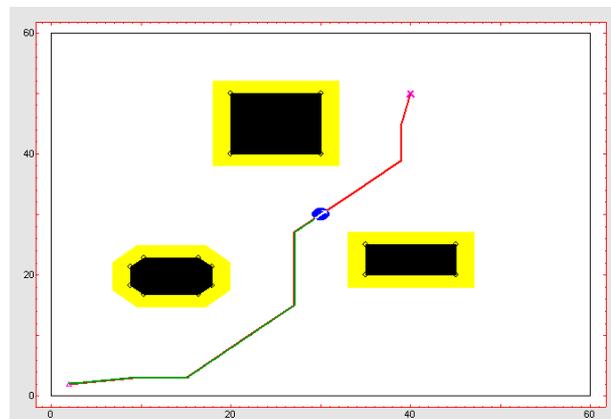
(a) Grafo de visibilidad.



(b) Diagrama de Voronoi.



(c) Quadtree.



(d) Frente de Onda.

Figura 3.20. Ejemplo de los algoritmos de planificación global.

Comparando los algoritmos de ambos tipos, globales y locales, el grafo de visibilidad es el algoritmo que calcula el camino más corto y más suave, mientras que el Diagrama de Voronoi consigue el camino más alejado de los obstáculos.

3.1.4. Estudio de persecución pura.

El algoritmo de *Persecución Pura* lo usan los algoritmos de planificación global para que el robot móvil siga la trayectoria calculada. Este algoritmo de seguimiento tiene como parámetros característicos el tiempo de muestreo T_m y la distancia al punto objetivo d .

Al variar el tiempo de muestreo, se consigue que el robot móvil disponga de más consignas para seguir la trayectoria planificada, por lo que ésta será más suave a menor tiempo de muestreo. En la figura 3.21 puede observarse un ejemplo con tiempos de muestreo de 0.1 y 0.03 segundos.

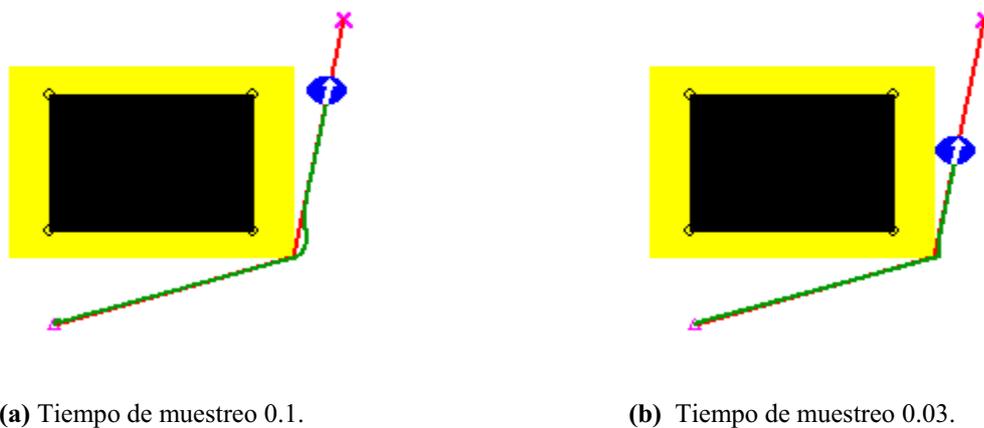


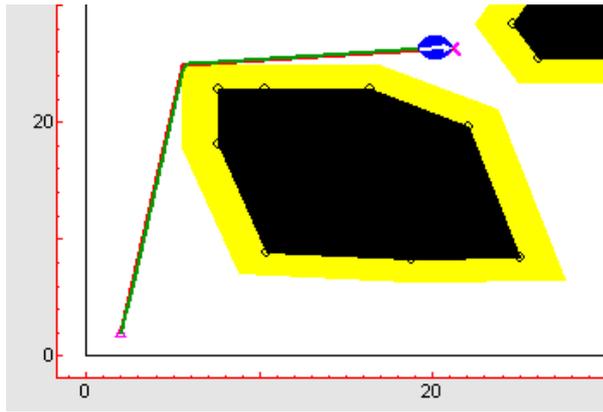
Figura 3.21. Ejemplo de persecución pura con distintos valores de tiempo de muestreo T_m .

El segundo parámetro característico, la distancia al punto objetivo d , hace que el robot móvil tome las curvas más cerradas o más abiertas. Como puede observarse en la figura 3.22, a menor valor de d menor suavidad a la hora de tomar las curvas, y a mayor valor, mayor arco de circunferencia y, por tanto, mayor suavidad.

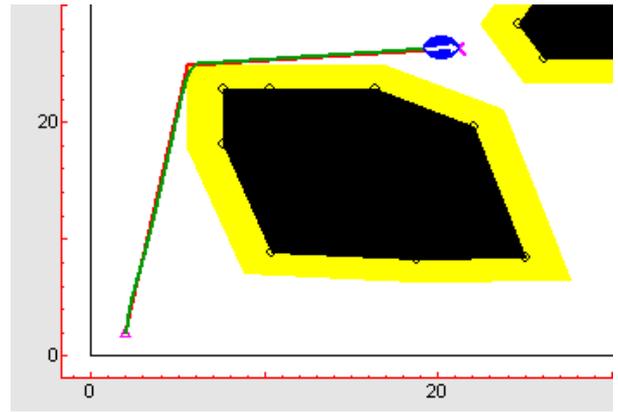
A la hora de seguir una trayectoria mediante *Persecución Pura*, el valor de los parámetros característicos del robot móvil también afecta a ésta. En la figura 3.23 se observa un ejemplo con distintas orientaciones iniciales, y en la figura 3.24 un ejemplo con donde se observa el efecto para distintas velocidades.

3.2. Comparación de comportamiento de cinemáticas.

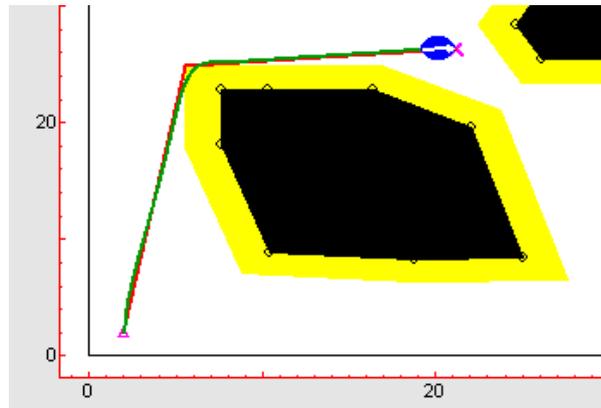
Las cinemáticas que incorpora la herramienta son la diferencial, triciclo y síncrona. Cada cinemática tiene señales de control características que definen su comportamiento. En las siguientes secciones se presentan varios ejemplos con el fin de observar el efecto que poseen las señales de control de las distintas cinemáticas.



(a) Distancia al punto objetivo 2.

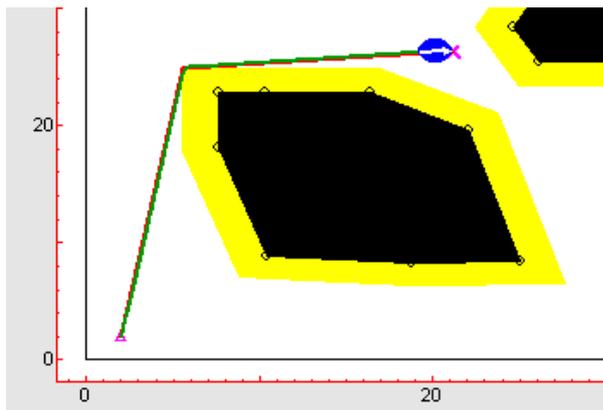


(b) Distancia al punto objetivo 5.

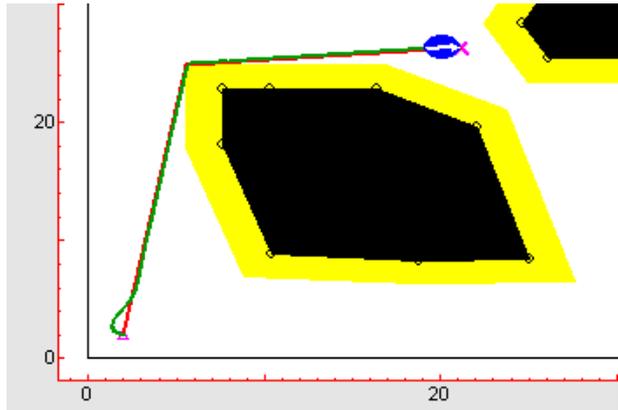


(c) Distancia al punto objetivo 8.

Figura 3.22. Ejemplo de persecución pura con distintos valores de distancia al punto objetivo d .

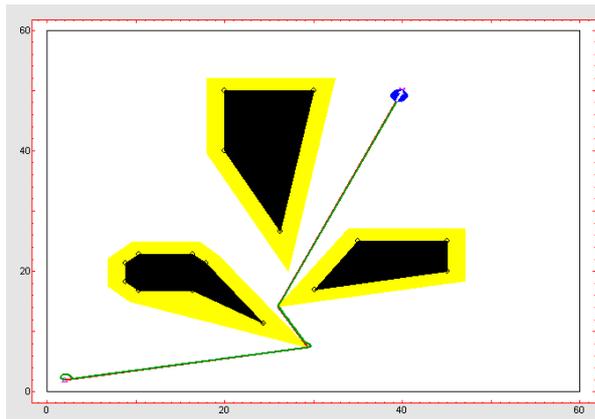


(a) Orientación inicial 1.57.

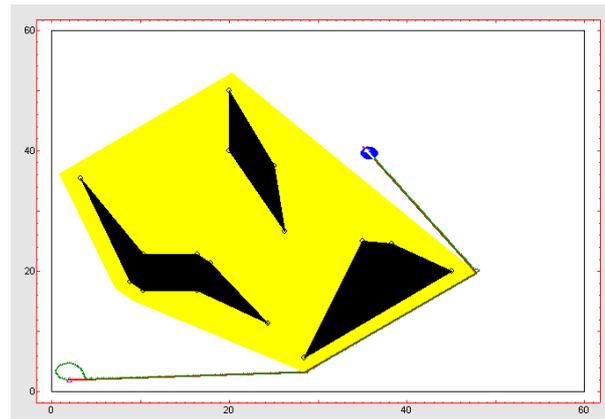


(b) Orientación inicial 2.6.

Figura 3.23. Ejemplo de persecución pura con distinta orientación inicial.



(a) Velocidad 1.

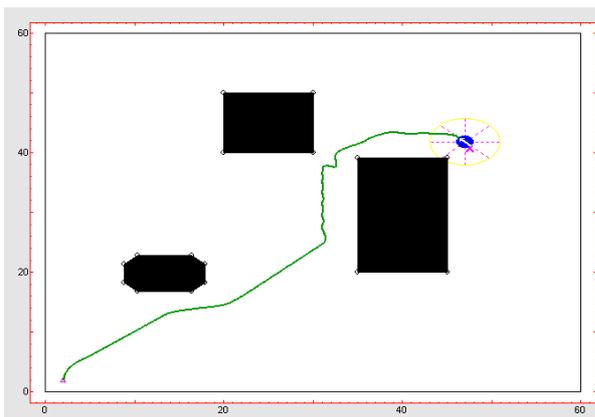


(b) Velocidad 2.

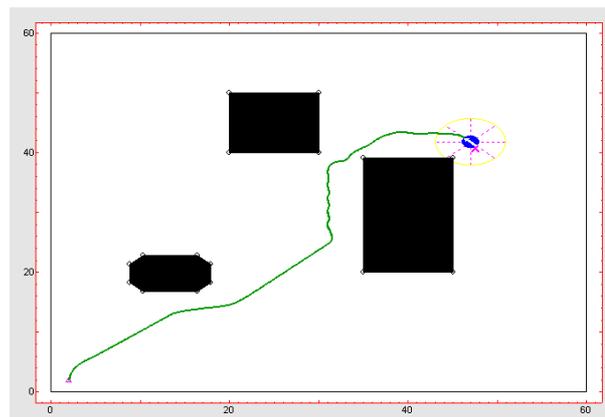
Figura 3.24. Ejemplo de persecución pura con distinta velocidad y la misma orientación inicial.

3.2.1. Ejemplos de las distintas cinemáticas.

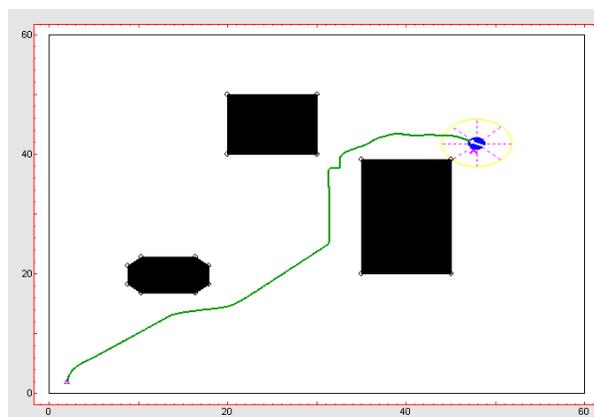
En la figura 3.25 se muestra un ejemplo donde se pueden observar los resultados del seguimiento de la misma trayectoria en un determinado entorno, haciendo uso de distintos modelos cinemáticos.



(a) Cinemática diferencial.



(b) Cinemática triciclo.



(c) Cinemática síncrona.

Figura 3.25. Algoritmo potencial con distintas cinemáticas.

En el ejemplo anterior se supone que el ángulo de giro de la rueda directriz del modelo triciclo está restringido entre 1 y -1 radianes, y el ángulo de orientación de las ruedas del modelo síncrono está restringido entre 1.5 y -1.5 radianes. Puede observarse como las curvaturas adoptadas por el robot móvil son distintas en el caso de cada cinemática, debido a las restricciones intrínsecas de las señales de control de cada uno de los modelos.

La figura 3.26 presenta un ejemplo más claro de comparación de cinemáticas donde la trayectoria planificada presenta un ángulo de noventa grados.

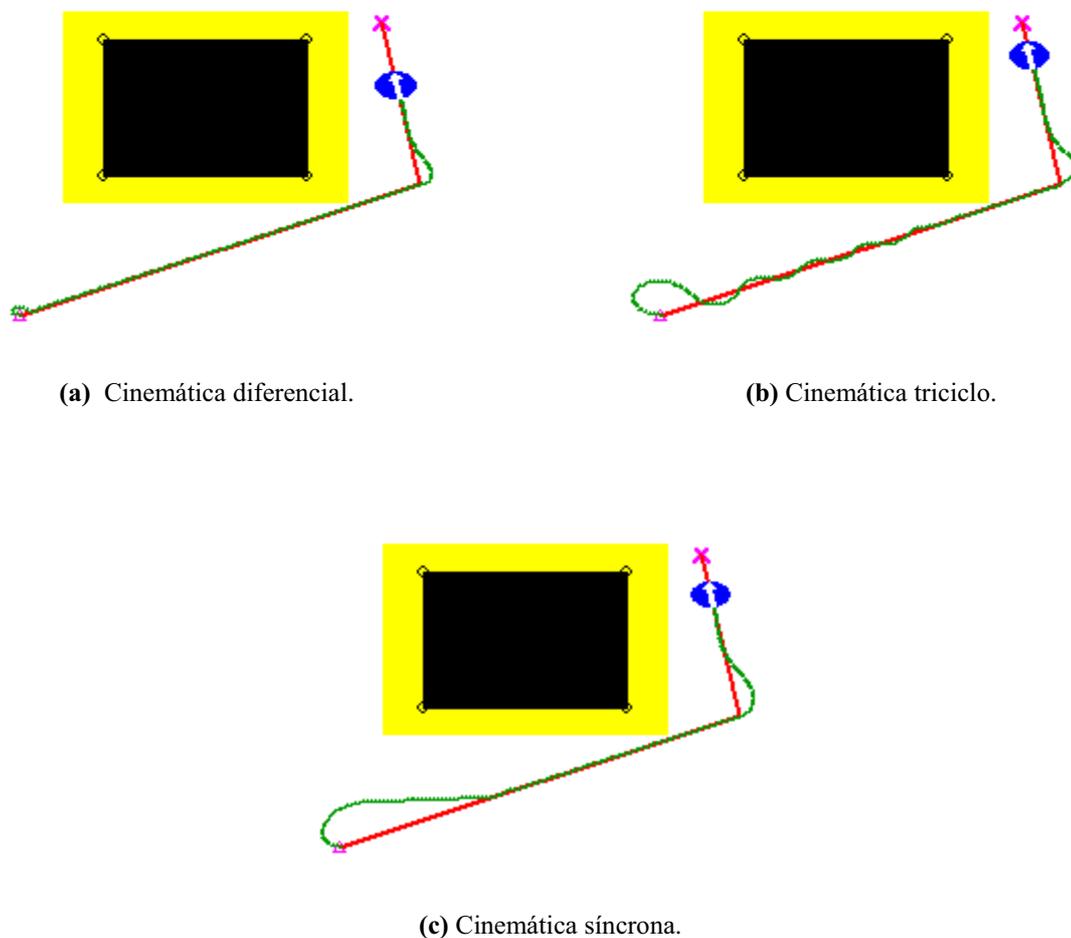
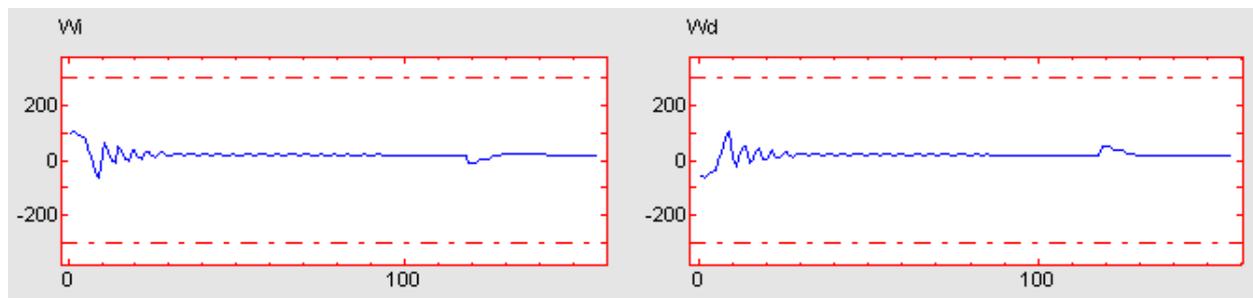


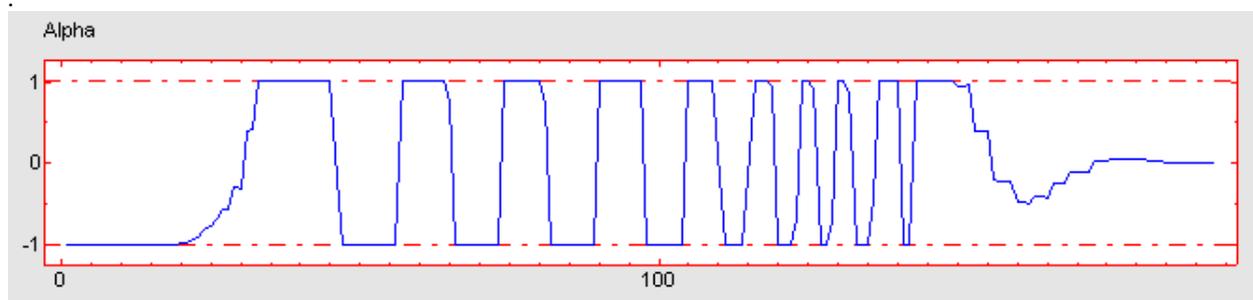
Figura 3.26. Algoritmo quadtree con distintas cinemáticas.

Nótese como la cinemática diferencial, la única de las tres con radio de giro nulo, minimiza los ángulos en los giros, sobre todo en la curva de noventa grados. La cinemática triciclo necesita un ángulo mayor tanto en la salida como en la curva, llegando incluso a zigzaguear al alcanzar la trayectoria planificada debido a la velocidad del robot móvil. Por último, la cinemática síncrona hace que el robot realice giros más lentos y arqueados, debidos a la limitación en la orientación de las ruedas del robot móvil.

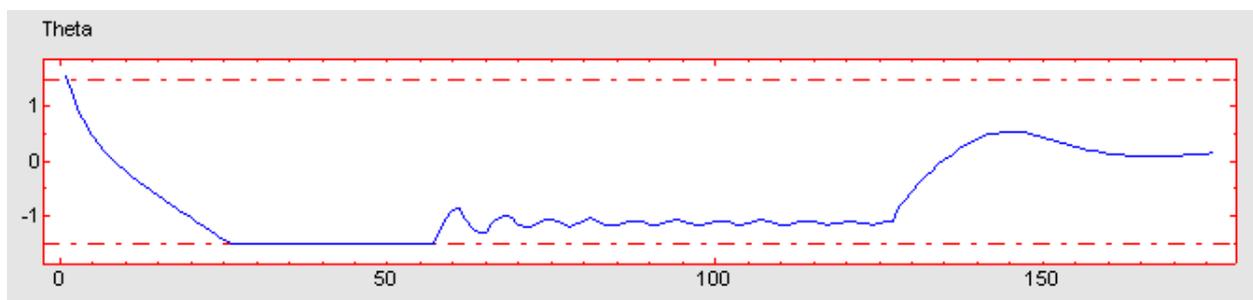
En la figura 3.27 pueden observarse las gráficas con los valores de las señales de control características de las cinemáticas de la figura 3.26.



(a) Señales de control de la cinemática diferencial.



(b) Señales de control de la cinemática triciclo.



(c) Señales de control de la cinemática síncrona.

Figura 3.27. Señales de control características de las distintas cinemáticas.

3.2.2. Estudio de restricciones.

Como ya se comentó anteriormente, las gráficas de la herramienta donde se muestran las señales de control para cada una de las cinemáticas, son gráficas interactivas. Esto quiere decir que se pueden modificar los límites superior e inferior de dichas señales de cada cinemática, simulando así, por ejemplo problemas de bloqueo en las ruedas del robot móvil.

En la figura 3.28 se muestra un ejemplo de robot móvil con cinemática diferencial donde se simula un defecto en la rueda derecha mediante una restricción en la señal de control ω_d . En la figura 3.29 se muestran las gráficas de las señales de control de la cinemática diferencial para dicho ejemplo.

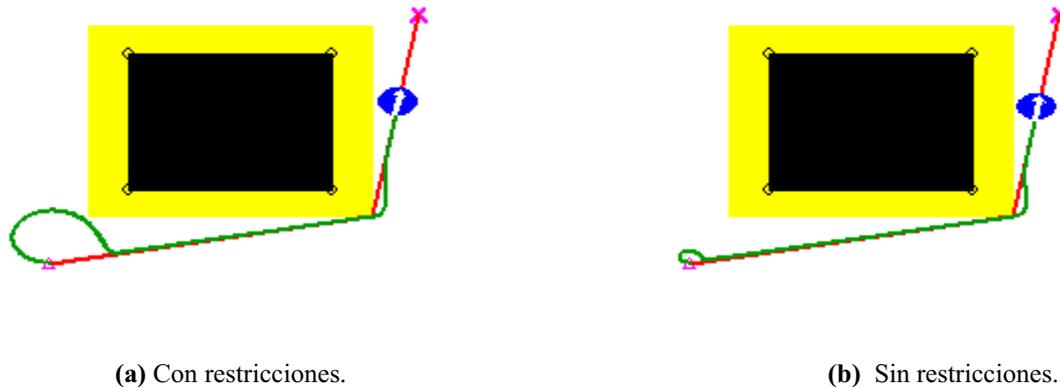


Figura 3.28. Ejemplo de cinemática diferencial con y sin restricciones.

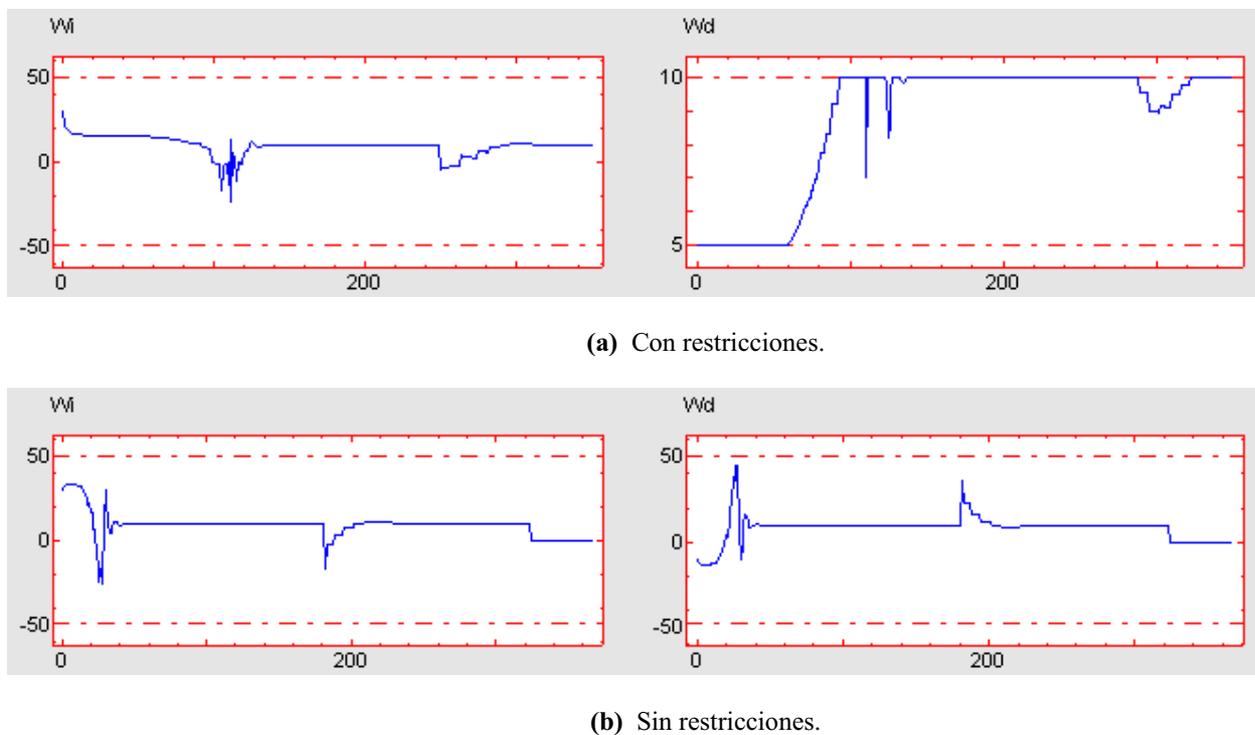


Figura 3.29. Señales de control características de la cinemática diferencial con y sin restricciones.

Puede observarse como este defecto simulado de la rueda derecha hace que la curvatura del robot móvil esté limitada y no le permita tomar curvas ligeramente cerradas, necesitando para ello más tiempo y un ángulo más abierto que en el caso sin restricciones.

En el caso de la cinemática del triciclo, su señal de control es el ángulo de giro de la rueda directriz (α). En la figura 3.30 se muestra un ejemplo de robot móvil con cinemática triciclo donde se simula un defecto en la rueda directriz mediante la modificación de la restricción base de la señal de control α . En la figura 3.31 se muestran las gráficas de la señal de control de la cinemática triciclo para dicho ejemplo.

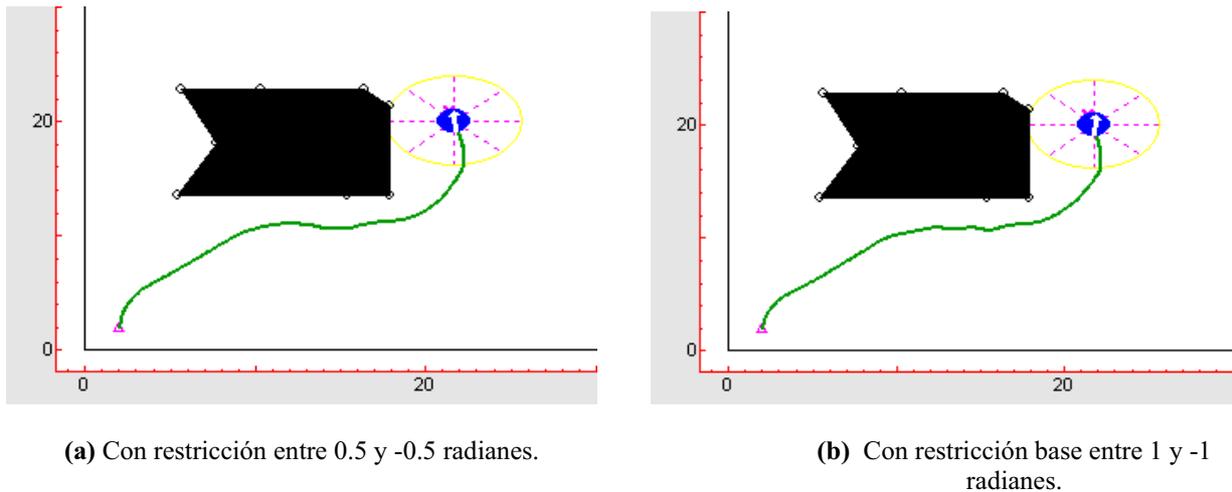


Figura 3.30. Ejemplo de cinemática triciclo con restricciones.

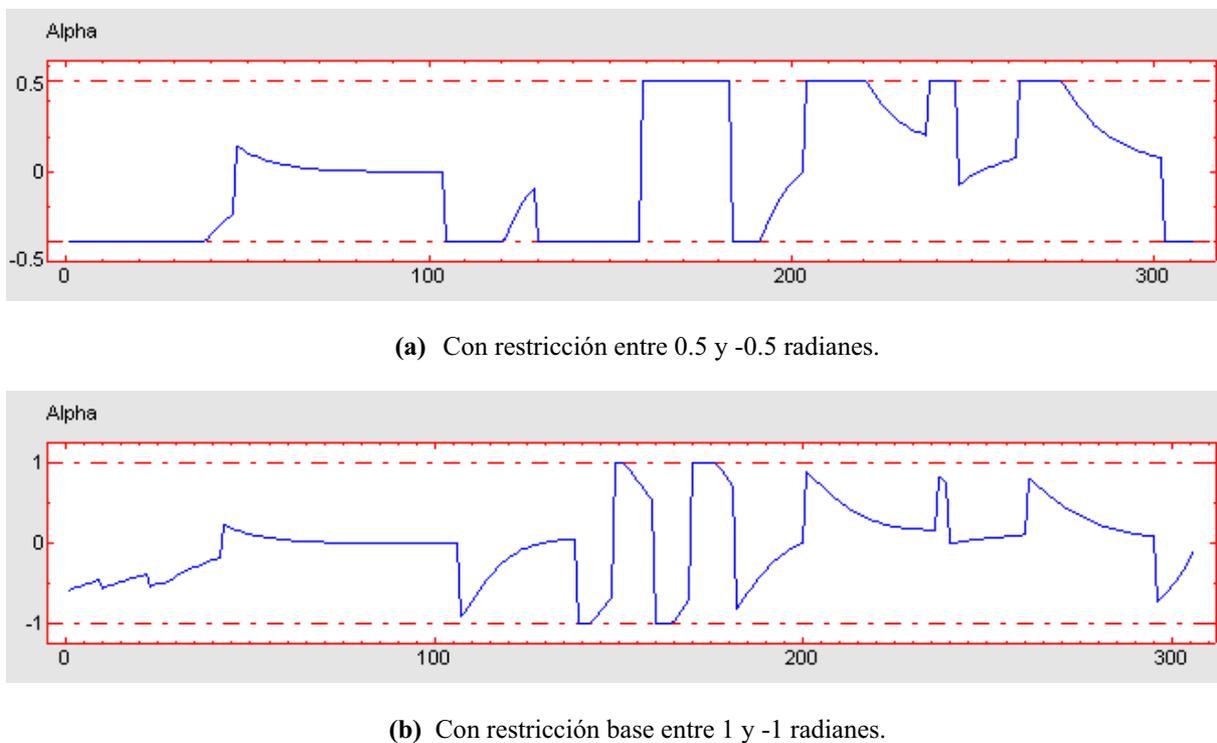


Figura 3.31. Señal de control característica de la cinemática triciclo con restricciones.

En las gráficas anteriores se observar como este problema simulado en la rueda directriz hace que la curvatura del robot móvil esté limitada y éste tenga que tomar curvas más abiertas de lo necesario.

La cinemática síncrona tiene como señal de control el ángulo ϕ de orientación de las ruedas. En la figura 3.32 se muestra un ejemplo de robot móvil con cinemática síncrona donde se simula un defecto en la orientación de las ruedas mediante la modificación de la restricción base de la señal de control ϕ . En la figura 3.33 se muestran las gráficas de la señal de control característica de la cinemática síncrona para dicho ejemplo.

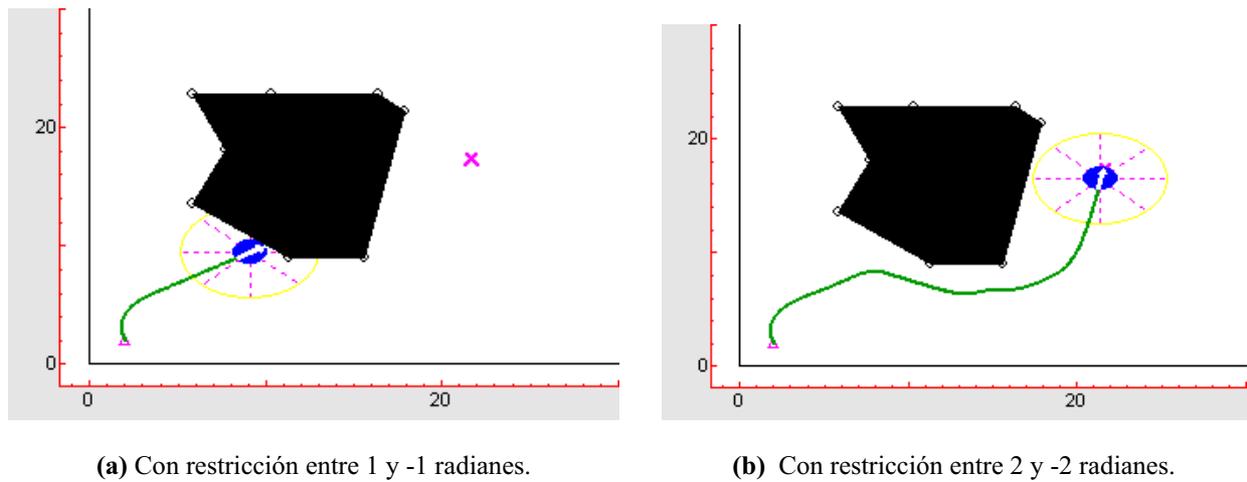


Figura 3.32. Ejemplo de cinemática síncrona con restricciones.

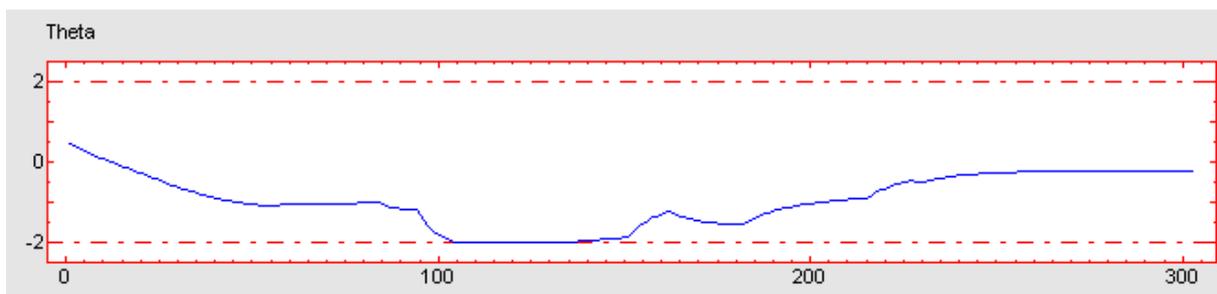
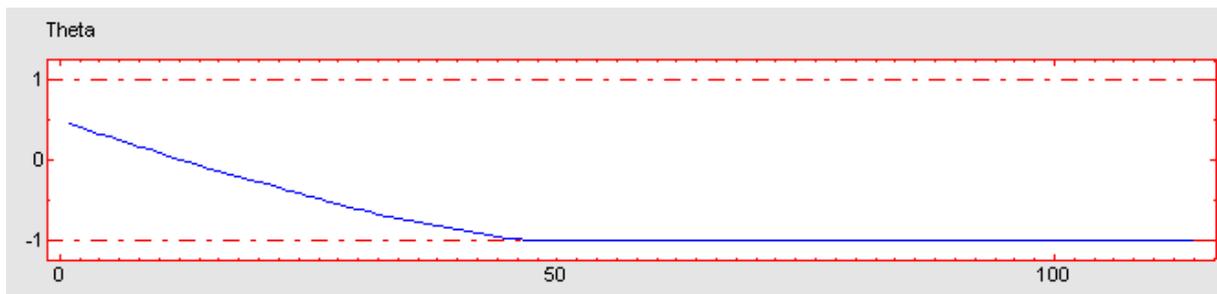


Figura 3.33. Señal de control característica de la cinemática síncrona con restricciones.

En los resultados obtenidos se observa como al restringir la orientación de las ruedas del robot éste no puede superar el obstáculo, colisionando con él debido a que las ruedas no pueden girar el ángulo necesario para evitarlo.