

**-TRABAJO FIN DE MÁSTER-
MÁSTER EN INGENIERÍA MATEMÁTICA**



**MODELIZACIÓN, IMPLEMENTACIÓN Y ANÁLISIS
DE UN ALGORITMO LOCAL DE NAVEGACIÓN EN
ENTORNOS DINÁMICOS**

Autor: Carlos Calvo Tapia

Tutor: Valeriy Makarov Slizneva



**Departamento de Matemática Aplicada
Facultad de Ciencias Matemáticas
Universidad Complutense de Madrid
Septiembre de 2014**

Índice general

1. Introducción	3
2. Algoritmo de navegación <i>Simple Rules</i> (SR)	5
2.1. Heurísticas del modelo	5
2.2. Efecto de posibles colisiones	8
3. Navegación en entornos sociales	9
3.1. Ejemplos de navegación usando SR	9
3.2. Análisis cuantitativo del modelo SR	12
4. Limitaciones cognitivas del modelo SR	17
4.1. Ejemplo de bajo nivel cognitivo del modelo SR	17
4.2. Limitaciones dependientes de la distancia del horizonte fijada	18
4.2.1. Resultado principal	21
4.2.2. Posibles direcciones iniciales del agente	21
4.2.3. Discusión de los mínimos locales de la función d	23
4.2.4. Discriminación de las zonas de fallo y acierto	27
4.2.5. Demostración del resultado principal	31
4.3. Limitaciones no dependientes de la distancia del horizonte fijada	31
4.3.1. Resultado principal	31
4.3.2. Construcción analítica de una subregión de fallo de- pendiente de la distancia del horizonte	32
4.3.3. Evolución de la subregión de fallo a medida que au- menta la distancia del horizonte	34
4.3.4. Demostración del resultado principal	35
5. Conclusiones	39
Bibliografía	40

Capítulo 1

Introducción

La naturaleza ha dotado a los seres vivos de un conjunto de capacidades cognitivas que les permite percibir el mundo externo de manera subjetiva, con el propósito de construir determinados comportamientos y sobrevivir en entornos específicos. Un ejemplo clásico es la destreza motora desarrollada por un agente móvil (ser humano, animal) para alcanzar un objetivo concreto en un entorno dinámico. Dicha destreza requiere la generación rápida y fiable de una decisión flexible que tenga en cuenta no sólo el estado actual del entorno y del agente, sino que también infiera sobre sus estados futuros [1].

Las habilidades cognitivas requeridas para la navegación en entornos dinámicos son muy básicas y han sido observadas incluso en animales de naturaleza muy simple [2]. Sin embargo, su descripción matemática e implementación en agentes artificiales (robots) continúa siendo un desafío. Un robot que haga frente de manera autónoma a este problema debe, por un lado, comprender el entorno en términos de “lo que puede pasar” y “cómo puedo afrontarlo”; y por otro, aprender de experiencias exitosas para hacer uso de ellas posteriormente de manera automática (“subconsciente”).

En las últimas décadas el estudio del fenómeno de la cognición se ha abordado desde múltiples disciplinas. La biología, la neurociencia, la robótica o la psicología han proporcionado perspectivas complementarias a este problema. Sin embargo, el conocimiento que se tiene acerca de los mecanismos implicados en los procesos cognitivos relacionados con los movimientos versátiles y eficaces que se observan en los seres vivos es muy limitado. Según la paradoja de Moravec, descubierta en los años 80 del siglo XX, el pensamiento razonado de alto nivel requiere de poca computación, mientras que las habilidades sensoriales y motoras de bajo nivel requieren de grandes esfuerzos computacionales.

En un estudio previo, [3], hemos propuesto abordar el problema de la navegación en entornos dinámicos mediante un modelo de red neuronal que simula las posibles interacciones entre un agente cognitivo y el entorno social

(en presencia de otros seres cognitivos) en que navega. El uso emergente de mapas cognitivos, en los que se compactifica la dimensión temporal de la situación dinámica afrontada, permite la planificación sencilla de rutas factibles que conducen al agente hasta su destino [4].

El trabajo mencionado proporciona un abordamiento global (tanto en espacio como en tiempo) del problema de la navegación cognitiva. La solución propuesta presenta considerables avances y múltiples ventajas; sin embargo, el sistema dinámico empleado para la generación de trayectorias es sumamente complejo.

En este trabajo se desarrollará el estudio de un modelo alternativo de navegación cognitiva, introducido por Moussaïd, Helbing y Theraulaz [5]. Dicho modelo se basa en dos heurísticas extraídas directamente del comportamiento humano. Las ecuaciones implicadas, pese a su relativa sencillez, permiten implementar, bajo reglas locales en tiempo, un sistema de navegación capaz de reproducir fenómenos de autoorganización en muchedumbres [6], como puede ser la formación espontánea de carriles bidireccionales en espacios en los que conviven individuos que desean desplazarse en sentidos opuestos [7].

El modelo matemático de navegación será implementado en Matlab y se analizarán de manera empírica y analítica los beneficios e inconvenientes de su uso. Además, se demostrará analíticamente que la localidad intrínseca del modelo limita significativamente su capacidad cognitiva.

Capítulo 2

Algoritmo de navegación *Simple Rules* (SR)

A lo largo de este capítulo se desarrollará una modelización matemática del sistema de navegación cognitiva propuesto por Moussaid, Helbing y Theraulaz [5]. El artículo citado propone el uso de dos sencillas reglas (*Simple Rules*, en inglés) a la hora de determinar las trayectorias que los peatones siguen cuando navegan entre muchedumbres. De aquí en adelante, se hará referencia a este modelo mediante sus siglas SR (*Simple Rules*).

2.1. Heurísticas del modelo

El modelo SR está basado en dos heurísticas del comportamiento humano. Considérense los movimientos de los peatones en un plano, tal y como muestra la Figura 2.1. Guiados por la información visual de su entorno, en concreto las distancias a los obstáculos que se interponen en cada una de las direcciones abarcables, los individuos emplean dos procedimientos cognitivos para adaptar a cada escenario concreto sus velocidades y direcciones de la manera más adecuada.

En el modelo SR, cada peatón $i \in I = \{1, 2, \dots, N\}$ queda caracterizado por su posición espacial en cada paso $p = 0, 1, 2, \dots$, que denotaremos $\vec{x}_i^p \in \mathbb{R}^2$. Por simplicidad, se asumirá que todos los individuos recalculan sus rutas en los mismos instantes, $t = \tau p$, donde τ es el tiempo entre dos pasos consecutivos; además, se considerará que todos ellos tienen un mismo campo de visión, el cual abarca un ángulo ϕ a izquierda y a derecha de su dirección visual en cada paso, \vec{H}_i^p . Asimismo, se asociará a cada uno un destino \vec{O}_i , una masa m_i y una velocidad confortable para sus aptitudes motoras, v_i° .

El área personal ocupada por cada individuo será modelizada como una circunferencia de radio r_i , proporcional a su masa, y en cada paso se tendrá en cuenta el ángulo $\alpha_{0,i}^p$ formado por la dirección de destino $\vec{O}_i - \vec{x}_i^p$ y la

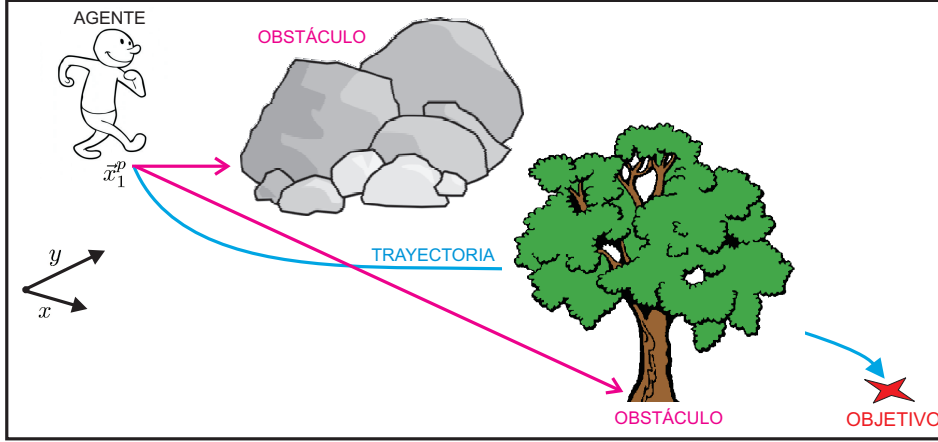


Figura 2.1: El agente cognitivo tiene que elegir una trayectoria esquivando los obstáculos que se encuentran al alcance de su vista. Se modelizarán escenarios como éste proyectando las localizaciones de agentes y obstáculos en el plano del suelo.

visual \vec{H}_i^p .

El algoritmo local propuesto responde esencialmente a dos heurísticas. La primera de ellas establece:

H₁: *Los peatones eligen la dirección $\tilde{\alpha}$ que les permite llegar de la manera más directa posible a su destino, teniendo en cuenta la presencia de obstáculos.*

Matemáticamente, definida una distancia D , que llamaremos *distancia del horizonte* (más allá de esta distancia el agente no alcanza a ver nada), se considerará, en cada paso p y para cada par de individuos $i, j \in I$, la función

$$f_{ij}^p : [-\phi, \phi] \longrightarrow [0, D],$$

que asigna a cada valor $\alpha \in [-\phi, \phi]$ el mínimo entre D y la distancia al choque con j previsto por i cuando considera tomar la dirección α .

Del mismo modo se define en cada paso p , para cada individuo $i \in I$ y cada pared $w \in W$, la función

$$f_{iw}^p : [-\phi, \phi] \longrightarrow [0, D],$$

que de manera análoga, asocia a cada valor α el mínimo entre D y la distancia al choque con w previsto por i cuando considera tomar la dirección α .

A partir de estas funciones auxiliares, se definen, para cada peatón $i \in I$ y en cada paso p , las funciones $f_i^p : [-\phi, \phi] \longrightarrow [0, D]$ (ver Figura 2.2) y

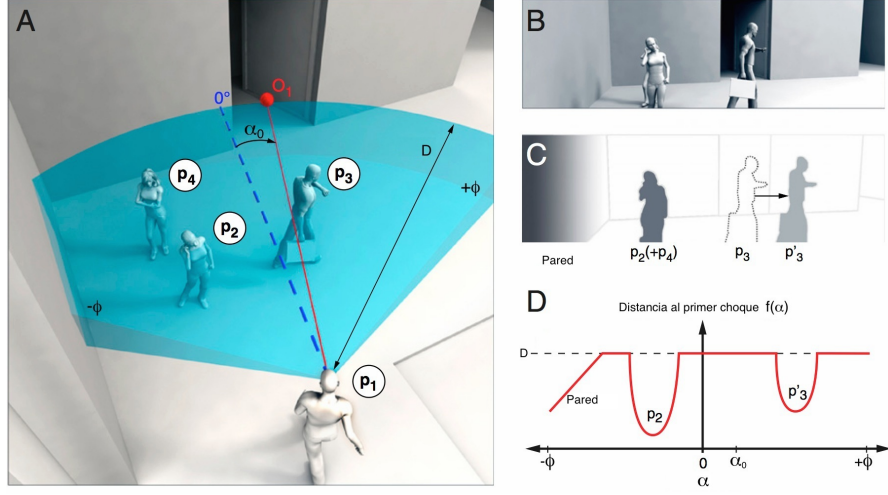


Figura 2.2: (A) Situación particular en la que el peatón p_1 , en interacción con otros tres individuos, desea alcanzar el punto O_1 (señalado en rojo); la recta discontinua azul representa su dirección visual, \vec{H}_1 . El área sombreada representa el campo visual de p_1 . (B) Visión que tiene p_1 de la situación anterior. (C) Abstracción de la situación dinámica afrontada; se ha empleado para ello un degradado visual en blanco y negro: las zonas más oscuras representan los choques más inminentes. (D) Gráfica de la función $f(\alpha)$, que representa la distancia a la que se produciría una colisión en la dirección α ; el lado izquierdo del campo de visión está limitado por una pared; p_4 no influye en la definición de $f(\alpha)$, por encontrarse oculto tras p_2 ; el individuo p_3 se encuentra en movimiento, por lo que el choque contra él se prevé en la dirección p'_3 . Figura adaptada de [5].

$d_i^p : [-\phi, \phi] \longrightarrow [0, +\infty)$ como sigue:

$$f_i^p(\alpha) = \min(\{f_{ij}^p(\alpha) : j \in I - \{i\}\} \cup \{f_{iw}^p(\alpha) : w \in W\})$$

$$d_i^p(\alpha) = D^2 + (f_i^p(\alpha))^2 - 2Df_i^p(\alpha) \cos(\alpha_{0,i}^p - \alpha)$$

La elección $\tilde{\alpha}_i$ a la que se refiere la primera heurística se realiza minimizando esta última función. En cada paso p :

$$\tilde{\alpha}_i^p = \arg \min\{d_i^p(\alpha) : \alpha \in [-\phi, \phi]\}$$

La segunda heurística establece:

H₂: *Los peatones mantienen en su camino una distancia prudente al primer obstáculo aparente en la dirección elegida; esta distancia al menos ha de asegurarle un tiempo de reacción cuanto menos igual al tiempo entre pasos, τ .*

Esto implica la elección de una velocidad adecuada bajo este criterio, que denotaremos \tilde{v} .

Por tanto, si se considera en cada paso p la distancia h_i^p existente entre i y el obstáculo más cercano a su ubicación, habrá de adaptar:

$$\tilde{v}_i^p = \min\{v_i^\circ, h_i^p/\tau\}$$

2.2. Efecto de posibles colisiones

Hasta ahora se ha establecido el modo en el que cada individuo $i \in I$ escoge, en cada paso, la dirección y la rapidez más convenientes en el camino a su punto de destino. Queda de este modo determinado su vector velocidad en cada paso p , \vec{v}_i^p , como aquél de módulo \tilde{v}_i^p y dirección $\tilde{\alpha}_i^p$. Resta tan sólo considerar el efecto de posibles colisiones entre elementos del sistema, que afectarían de manera sustancial a su dinámica.

Sea $g : \mathbb{R} \rightarrow [0, +\infty)$, definida como

$$g(u) = u\mathbf{H}(u),$$

donde $\mathbf{H} : \mathbb{R} \rightarrow \{0, 1\}$ representa la función *Heaviside*. Llamando d_{ij}^p a la distancia en el paso p entre los individuos i y j , i.e. $d_{ij}^p = \|\vec{x}_i^p - \vec{x}_j^p\|$, y considerando el vector unitario $\vec{n}_{ij}^p = \frac{1}{d_{ij}^p} (\vec{x}_i^p - \vec{x}_j^p)$, la fuerza ejercida sobre i por el efecto de una posible colisión con j viene dada por la expresión

$$\vec{f}_{ij}^p = Kg(r_i + r_j - d_{ij})\vec{n}_{ij}^p,$$

donde $K > 0$ es una constante y g es la función real auxiliar arriba definida.

Análogamente, considerando la mínima distancia d_{iw}^p desde la ubicación del agente i a la pared $w \in W$ y definiendo \vec{n}_{iw}^p como el vector unitario en la dirección perpendicular a dicha pared (apuntando hacia el interior del habitáculo), la fuerza ejercida sobre i por el efecto de una posible colisión con w viene dada por la expresión

$$\vec{f}_{iw}^p = Kg(r_i - d_{iw})\vec{n}_{iw}^p.$$

Basta entonces integrar el problema de valor inicial

$$\begin{cases} \dot{\vec{x}}_i(z) = \vec{v}_i(z) \\ \dot{\vec{v}}_i(z) = \frac{1}{\tau} (\vec{v}_i^p - \vec{v}_i(z)) + \frac{1}{m_i} \left(\sum_{j \in I - \{i\}} \vec{f}_{ij}^p + \sum_{w \in W} \vec{f}_{iw}^p \right) \\ \vec{x}_i(p\tau) = \vec{x}_i^p \\ \vec{v}_i(p\tau) = \vec{v}_i^p \end{cases} \quad z \in [p\tau, (p+1)\tau]$$

para obtener la posición de cada peatón $i \in I$ en el paso $p + 1$:

$$\vec{x}_i^{p+1} = \vec{x}_i((p+1)\tau).$$

Capítulo 3

Navegación en entornos sociales

Pese a tratarse de un modelo más simple que otros muchos del estado del arte, SR es capaz de predecir trayectorias individuales y patrones colectivos de movimiento coherentes con gran variedad de observaciones empíricas y experimentales. Para poder realizar experimentos numéricos se ha programado en Matlab el modelo SR descrito en el Capítulo 2 y se han simulado los movimientos de peatones en diferentes entornos. Se han empleado, en todos los casos, los siguientes parámetros: $\phi = 75^\circ$; $\tau = 0.5$; $D = 10$; $K = 5000$.

3.1. Ejemplos de navegación usando SR

Para probar la eficacia del modelo SR a la hora de generar trayectorias en diferentes situaciones reales vamos a considerar tres escenarios distintos.

Escenario 3.1 (Figura 3.1.A). *Un único agente interviene en este caso, pretendiendo alcanzar un objetivo estático mientras se desplaza entre cuatro bloques rectangulares que obstruyen su camino.*

La Figura 3.1.B representa la trayectoria del agente (se han sobrepuesto fotogramas para poder visualizar el movimiento descrito). Puede contemplarse que, en efecto, la situación es resuelta de manera exitosa y el agente consigue alcanzar su destino sin experimentar colisión alguna.

En el segundo escenario propuesto desaparecen los obstáculos estáticos. A cambio, el número de individuos que intervienen asciende a cuatro.

Escenario 3.2 (Figura 3.2.A). *Cuatro peatones navegan en una arena común. La complicación de esta situación particular radica en que la decisión que cada uno de ellos desearía tomar en caso de encontrarse aislados implicaría pasar por una zona común: el centro de la escena.*

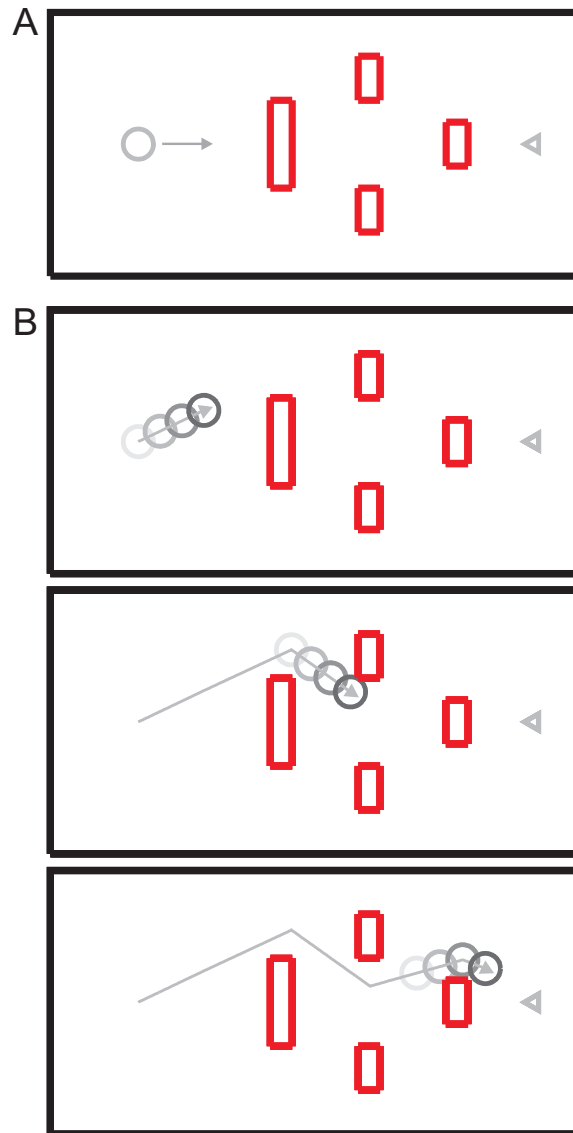


Figura 3.1: Navegación SR en un entorno estático (Escenario 3.1). (A) Situación inicial. El agente (su área personal) es representado mediante una circunferencia gris. La flecha indica su dirección visual inicial. El triángulo señala el destino que desea alcanzar. Los bloques rojos representan muros sólidos. (B) Solución proporcionada por el sistema de navegación SR. Los tres paneles se corresponden con tres instantes de tiempo consecutivos. El incremento de intensidad en la escala de grises empleada corresponde con las posiciones progresivas del agente en los sucesivos pasos.

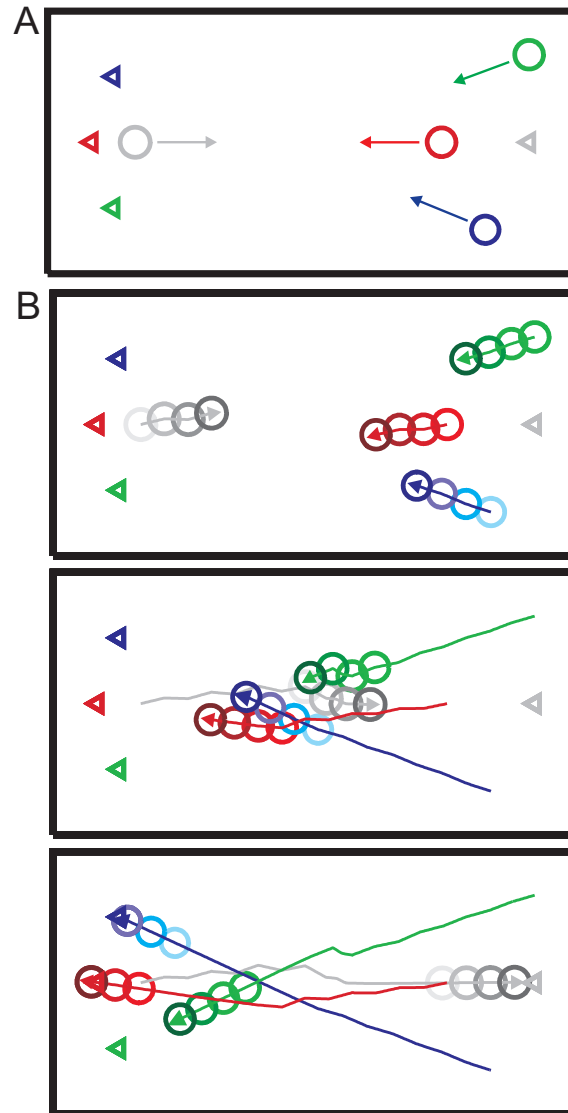


Figura 3.2: Navegación SR en un entorno social: cuatro agentes han de cooperar para alcanzar sus metas (Escenario 3.2). (A) Situación inicial. Los agentes (sus áreas personales) son representados mediante circunferencias de diferentes colores. Las flechas indican sus respectivas direcciones visuales iniciales. Cada agente desea alcanzar el triángulo del color asociado. (B) Solución proporcionada por el sistema de navegación SR. El incremento de intensidad en la escala de colores empleada para cada individuo corresponde con sus posiciones progresivas en los sucesivos pasos.

Al igual que en la situación estática, el modelo SR es capaz de solventar de nuevo con éxito la situación descrita, tal y como se observa en la Figura 3.2.B.

El tercer escenario propuesto pretende mostrar que el modelo es capaz también de predecir un patrón de movimiento colectivo habitual como es la organización de una multitud en carriles unidireccionales cuando, en un pasillo alargado, individuos que desean alcanzar un extremo se encuentran con individuos que desean alcanzar el otro.

Escenario 3.3 (Figura 3.3, panel t_0). *Se han distribuido de manera aleatoria un total de sesenta agentes a lo largo de un pasillo alargado, treinta de ellos pretendiendo desplazarse en un sentido y treinta pretendiendo desplazarse en el otro.*

La Figura 3.3 (paneles t_1 , t_2 y t_3) muestra la distribución de los peatones en tres instantes consecutivos. Se puede observar cómo, pasado cierto tiempo, los agentes de uno y otro tipo deciden separar sus trayectorias y caminar de manera organizada, en dos filas.

3.2. Análisis cuantitativo del modelo SR

Una consecuencia natural de navegar en contextos de aglomeración es la necesidad casi permanente de alterar las rutas. Cuando se navega en entornos con una elevada concentración de individuos, mantener trayectorias de mínimo recorrido resulta inviable. En consecuencia, el coste de alcanzar un determinado objetivo (en términos de energía consumida) puede crecer de manera sustancial.

Planteemos una situación en la que un agente desea cruzar una sala en presencia de individuos que se oponen en su camino. Parece razonable pensar que el trayecto recorrido por el agente hasta alcanzar su destino será más largo cuanto mayor sea la densidad de individuos que caminen en sentido contrario. Es este hecho el que se propone demostrar de manera cuantitativa en esta sección.

Escenario 3.4 (Figura 3.4). *Considérese un agente inmerso en una escena coordinada: el espacio rectangular de expresión $(0, 5) \times (0, 2)$. Teniendo en cuenta que existen muros sólidos que delimitan este recinto a lo largo pero no a lo ancho, un agente desea desplazarse desde la posición inicial $\vec{x}_0 = (0, 1)$ hasta el destino $\vec{O} = (5, 1)$, mientras que de frente caminan otros agentes en sentido contrario.*

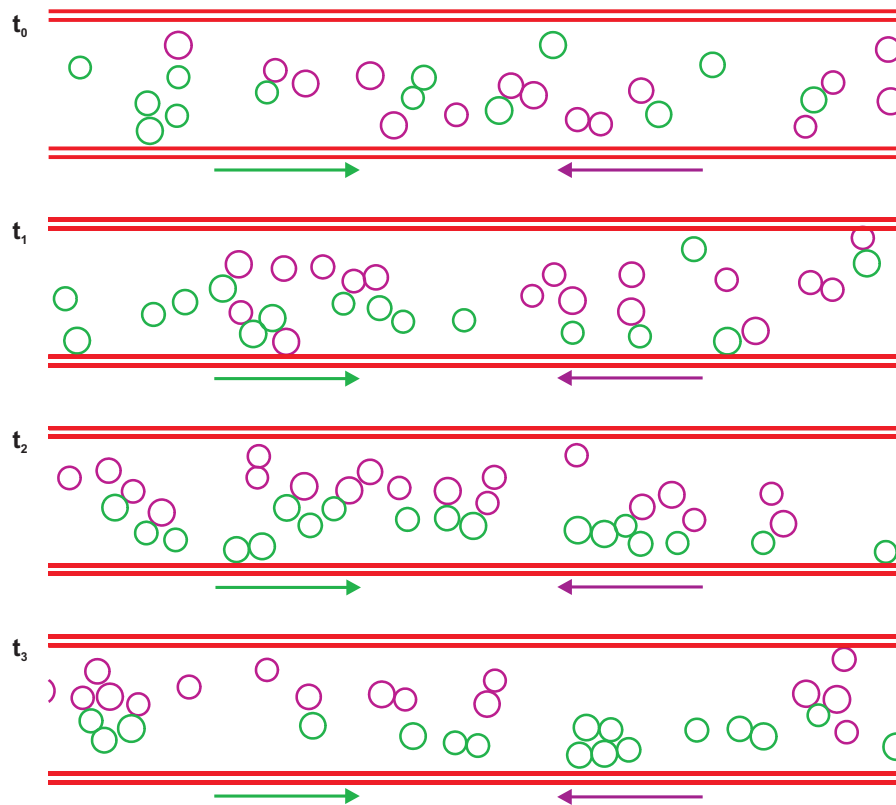


Figura 3.3: Formación emergente de dos filas de peatones que se cruzan en un pasillo siguiendo sentidos opuestos (Escenario 3.3). Solución proporcionada por el sistema de navegación SR. Panel t_0 : distribución inicial aleatoria. Los agentes representados en color verde desean dirigirse hacia la derecha de la imagen y los representados en magenta hacia la izquierda, tal y como indican los sentidos de las flechas. La figura muestra un segmento central del pasillo. Paneles t_1 - t_3 : distribución de la muchedumbre en los instantes $t_1 = 10$, $t_2 = 20$ y $t_3 = 30$. En las dos últimas capturas se observa una clara tendencia a distribuir las dos clases de trayectoria en dos carriles.

Cuando en la escena interviene sólo el citado agente, la solución proporcionada por el sistema de navegación SR es obviamente la trayectoria recta, de longitud mínima $L_0 = 5$, que resulta al unir mediante un segmento estos dos puntos.

Ahora bien, si se introducen en escena un total de N individuos, todos ellos caminando en oposición a la trayectoria obvia del agente anterior, la solución que proporcionará el algoritmo asociado al modelo SR incluirá muy posiblemente rodeos o incluso choques de unos individuos con otros, lo que podrá inducir un aumento considerable en la longitud de la trayectoria que finalmente guiará al agente inicial hasta su destino. La figura 3.4 muestra tres posibles disposiciones iniciales para una elección creciente de N .

Con el propósito de corroborar que el algoritmo es coherente con el hecho que se desea probar, se ha realizado, de acuerdo a las condiciones establecidas en el Escenario 3.4, el siguiente experimento: para cada posible valor del parámetro $N \in \{1, 2, \dots, 10\}$ se han extendido inicialmente y de manera aleatoria un total de N individuos sobre el ala derecha del recinto estudiado, todos ellos con intención de alcanzar un punto lejano a la izquierda del pasillo constituido entre los dos muros, en oposición al camino que desearía tomar el agente de posición inicial $\vec{x}_0 = (0, 1)$.

Las 15 simulaciones que se han realizado para cada posible valor de N han devuelto diferentes soluciones, habilitando en todos los casos al agente para que alcanzase su destino: $\vec{O} = (5, 1)$. La longitud L de cada una de estas trayectorias se ha comparado con la longitud base L_0 , correspondiente a la trayectoria trivial que se genera al fijar $N = 0$.

La Tabla 3.1 recoge, para cada repetición del experimento descrito, el cociente L/L_0 obtenido en función de la densidad de oponentes en el habitáculo, así como el valor medio y la desviación típica de dicho cociente para cada nivel de aglomeración. Es reseñable que, para altas concentraciones, se han generado ciertas disposiciones cuya solución ha llegado incluso a duplicar la longitud recorrida por el agente en su viaje hasta el objetivo.

La Figura 3.5 muestra de manera gráfica cómo el valor medio de la razón L/L_0 aumenta a medida que lo hace la cantidad de individuos que se oponen al agente principal. La variabilidad de esta razón también tiene tendencia creciente: a medida que se reducen los espacios entre los que el agente puede abrirse camino, las colisiones entre elementos del sistema aumentan y la aleatoriedad de las soluciones obtenidas induce resultados cada vez más variables.

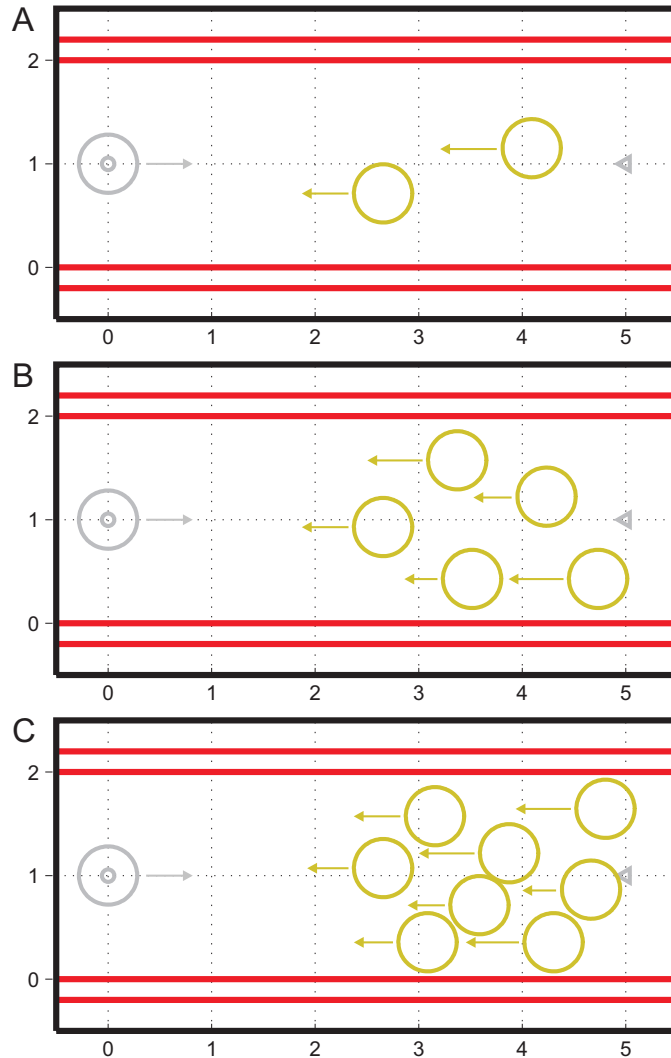


Figura 3.4: Tres posibles disposiciones iniciales recogidas en el Escenario 3.4. En todos los casos el agente principal, representado mediante una circunferencia de color gris, parte de la posición $\vec{x}_0 = (0, 1)$, con intención de alcanzar el objetivo $\vec{O} = (5, 1)$, marcado con un triángulo del mismo color. Las posiciones iniciales de los oponentes, representados en tono dorado, se han dispuesto de manera aleatoria, siguiendo una distribución uniforme sobre el subespacio rectangular $(2.5, 5) \times (0.5, 1.5)$. Cada flecha está dirigida en la dirección visual inicial de cada individuo; además, su longitud es proporcional a la *rapidez comfortable* v° de cada uno de ellos, que en todos los casos se ha escogido de manera aleatoria siguiendo una distribución uniforme sobre el intervalo $(0.83, 1.17)$. Se ha fijado la misma masa para todos los individuos que intervienen en la escena, de modo que las áreas personales abarcadas sean también las mismas en todos los casos. (A) $N = 2$. (B) $N = 5$. (C) $N = 8$.

		Oponentes por unidad cuadrada de espacio									
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Repeticiones	1	1.00	1.18	1.04	1.14	1.21	1.89	1.38	1.62	1.46	1.70
	2	1.00	0.97	1.06	1.18	1.39	1.26	1.87	1.94	2.02	1.80
	3	1.00	1.00	1.16	1.07	1.16	1.21	1.05	1.20	1.48	1.62
	4	1.00	1.25	1.13	1.56	1.39	1.08	1.07	1.25	1.25	1.51
	5	1.00	0.99	1.19	1.07	1.14	1.22	1.19	1.44	1.37	1.49
	6	1.00	0.98	1.16	1.61	1.26	1.22	1.53	1.43	2.18	1.27
	7	1.00	1.07	1.00	1.17	1.14	1.30	1.30	1.88	1.64	2.11
	8	1.00	0.99	1.07	1.18	1.65	1.25	1.22	1.62	1.73	2.07
	9	1.00	1.07	1.05	1.13	1.15	1.13	1.58	1.11	1.47	1.88
	10	1.00	1.00	1.51	1.23	1.29	1.59	1.18	1.06	1.37	1.61
	11	1.00	1.15	1.08	0.98	1.04	1.16	1.35	1.29	1.65	1.47
	12	1.00	1.21	1.24	1.23	1.14	1.30	1.19	1.61	1.26	2.30
	13	1.00	0.99	1.42	1.08	1.11	1.29	1.32	1.57	1.08	1.21
	14	1.00	1.09	1.13	1.07	1.12	1.02	1.30	1.40	1.50	1.31
	15	1.00	1.08	1.14	1.26	1.18	1.05	1.47	1.46	1.70	2.36
Media	1.00	1.07	1.16	1.20	1.22	1.26	1.33	1.46	1.54	1.71	
SDT	0.00	0.09	0.14	0.17	0.15	0.21	0.21	0.25	0.28	0.35	

Tabla 3.1: Valores del cociente L/L_0 obtenidos para cada nivel de aglomeración, en cada una de las 15 repeticiones simuladas bajo las condiciones expuestas en el Escenario 3.4. Las dos últimas filas recogen la media y desviación típica obtenidas para cada nivel, cuya representación gráfica puede observarse en la Figura 3.5.

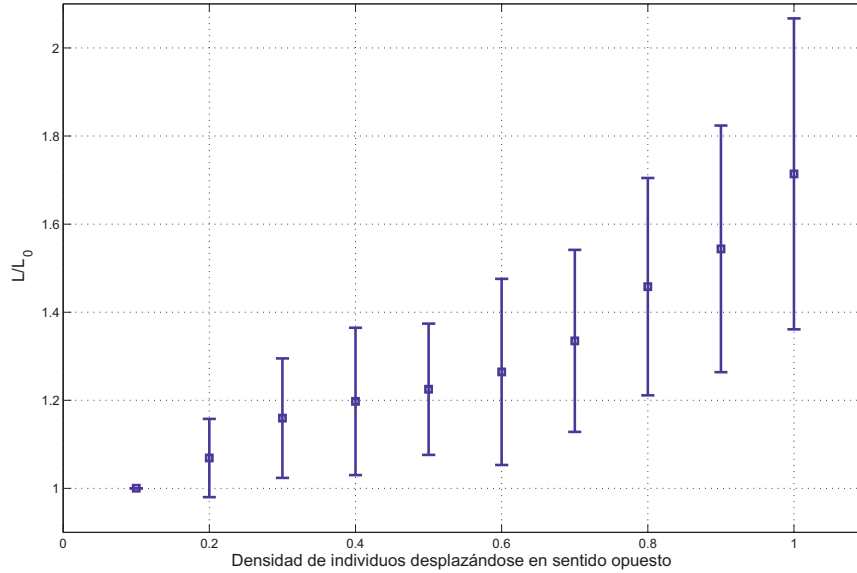


Figura 3.5: Rendimiento del modelo SR en entornos sociales con distinta densidad de peatones. La gráfica representa los valores medios y desviaciones típicas del cociente L/L_0 obtenido bajo las condiciones del Escenario 3.4, en función de la densidad de individuos que se oponen al trayecto del agente principal. Se han realizado 15 simulaciones para cada nivel de aglomeración. Los valores numéricos de las muestras obtenidas pueden observarse en la Tabla 3.1.

Capítulo 4

Limitaciones cognitivas del modelo SR

Tal y como se ha probado en la sección anterior, el modelo SR se comporta de manera razonable ante diversidad de situaciones planteadas, en coherencia además con estudios empíricos y experimentales en el campo de la psicología del comportamiento humano.

Sin embargo, la estructura del algoritmo es local en el tiempo; es decir, el algoritmo debe recalcular completamente las trayectorias cada paso p . El modelo SR, como ya se ha visto, se fundamenta en permanentes tomas de decisión que satisfacen dos heurísticas conductuales básicas. Pero estos dos criterios se satisfacen a muy corto plazo, y por tanto la solución final no tiene por qué corresponderse con una decisión global deseable. En cambio, los algoritmos globales generan trayectorias sólo una vez, lo que puede suponer un gran beneficio [2], [4].

4.1. Ejemplo de bajo nivel cognitivo del modelo SR

Para demostrar posibles problemas del modelo SR, derivados de su carácter local, consideremos la siguiente situación:

Escenario 4.1 (Figura 4.1.A). *Un individuo quiere alcanzar un objetivo que se encuentra tras un muro infinito que se extiende a lo largo de toda una semirrecta.*

En esta ocasión el ángulo de visión se ha fijado en $\phi = 90^\circ$ y se ha tomado un valor $D = 10$ para la distancia al horizonte. El muro se extiende hacia la izquierda de la imagen, más allá de los límites visuales del agente. En condiciones de normalidad, cualquiera capaz de comprender esta sencilla situación estática debería dirigir sus pasos hacia el borde inmediato del muro, a sabiendas de que su destino se encuentra tras él. Sin embargo, en esta ocasión el modelo denota un nivel cognitivo bajo y no encuentra vía libre, optando por lanzar al agente contra la pared (Figura 4.1.B).

Con el propósito de dar una explicación a este tipo de fallos en el modelo, a continuación se analizará una familia de escenarios para los que siempre es posible encontrar una condición inicial dada la cual el algoritmo de resolución expuesto devuelve un comportamiento absurdo del sistema.

4.2. Limitaciones dependientes de la distancia del horizonte fijada

Con el fin de demostrar analíticamente el resultado numérico presentado en la Sección 4.1, consideremos el siguiente escenario:

Escenario 4.2. *Un agente de masa m , radio r , amplitud visual ϕ y distancia del horizonte $D > 0$ desea alcanzar un target puntual estático de coordenadas $(x_T, y_T) \in (-D, 0) \times (0, D)$, evitando chocar con la pared semirrectilínea de grosor despreciable $w = (-\infty, -r) \times \{0\}$. Establecida en cualquier caso su dirección visual inicial en la dirección $\vec{H} = (0, 1)$, se desea conocer la dirección inicial $\tilde{\alpha}$ proporcionada por el modelo SR cuando el agente ocupa una posición inicial $(x, y) \in \mathbb{R} \times (-\infty, 0)$ desde la cual es capaz de observar vía libre para alcanzar el otro lado de la pared w , i.e.*

$$(x, y) \in A_D = \{(u, v) \in \mathbb{R} \times (-\infty, 0) : \|(u, v)\| < D, v \leq \cot(\phi)u\} \subset \mathbb{R}^2.$$

Por comodidad, se reducirá el agente a un punto, extendiendo los límites del obstáculo en la misma proporción: $W = (-\infty, 0) \times \{0\}$ (seguirá despreciándose su grosor). Y por simplicidad, se fijará una amplitud visual $\phi = 90^\circ$, de modo que

$$A_D = \{(u, v) \in \mathbb{R} \times (-\infty, 0) : \|(u, v)\| < D\} \subset \mathbb{R}^2.$$

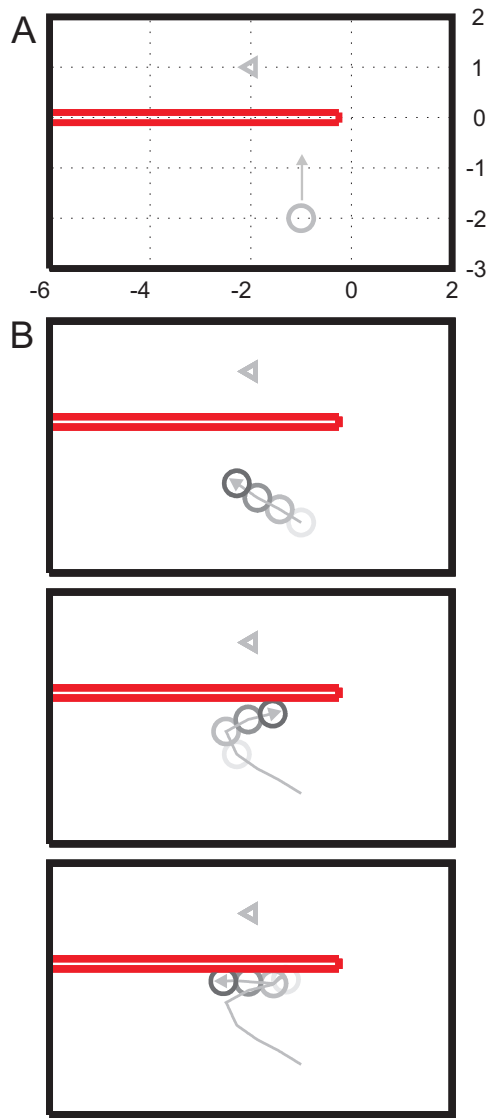


Figura 4.1: Ejemplo que demuestra un bajo nivel cognitivo del modelo SR (Escenario 4.1). (A) Situación inicial. La posición inicial del agente ha sido representada mediante una circunferencia gris de centro $(x_A, y_A) = (-1, -2)$ y radio $r = 0.2$. La flecha indica su dirección visual inicial, $\vec{H} = (0, 1)$. El triángulo señala el destino que desea alcanzar, de coordenadas $(x_T, y_T) = (-2, 1)$. El bloque rojo representa un muro sólido de grosor despreciable: $w = (-\infty, -r) \times \{0\}$. (B) Solución proporcionada por el sistema de navegación SR. El incremento de intensidad en la escala de grises empleada corresponde con las posiciones progresivas del agente en los sucesivos pasos. El agente no es capaz de comprender el escenario y no llega a alcanzar su meta. Se han empleado los siguientes parámetros: $\phi = 90^\circ$, $\tau = 0.5$, $D = 10$, $K = 5000$.

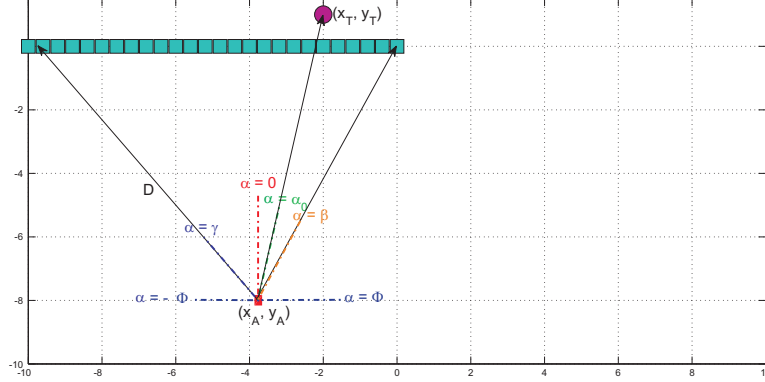


Figura 4.2: Representación de las direcciones $\pm\phi$, γ , α_0 y β para una localización inicial del agente (x_A, y_A) , una distancia del horizonte D y una posición del *target* (x_T, y_T) .

Una vez considerados todos estos supuestos, introducimos:

Definición 4.1. Para cada posible posición inicial del agente (x, y) se definen los siguientes ángulos (ver Figura 4.2), medidos con respecto a la dirección visual \vec{H} :

- $\alpha_0 = \arctan\left(\frac{x - x_T}{y - y_T}\right)$: representa la dirección al target.
- $\beta = \arctan\left(\frac{x}{y}\right)$: representa la dirección al borde de la pared W .
- $\gamma = -\arccos\left(-\frac{y}{D}\right)$: representa la dirección al punto más lejano de W alcanzable por la visión limitada del agente.

Es obvio que, en A_D , la dirección γ al punto más lejano de W está unívocamente determinada y que $\gamma < \beta$.

Los ángulos α_0 , β y γ serán de gran utilidad en el estudio analítico de la primera fase del algoritmo de resolución del modelo SR. Para ellos, se tiene:

$$\begin{aligned} \cos \alpha_0 &= \frac{y_T - y}{\|(x_T - x, y_T - y)\|} & \sin \alpha_0 &= \frac{x_T - x}{\|(x_T - x, y_T - y)\|} \\ \cos \beta &= -\frac{y}{\|(x, y)\|} & \sin \beta &= -\frac{x}{\|(x, y)\|} \\ \sin \gamma &= -\sqrt{1 - \left(\frac{y}{D}\right)^2} & \tan \gamma &= -\sqrt{\left(\frac{D}{y}\right)^2 - 1} \end{aligned}$$

4.2.1. Resultado principal

Analizando el Escenario 4.2 podemos formular el siguiente teorema:

Teorema 4.1. *Bajo las condiciones del Escenario 4.2, para cualquier valor $D > 0$, existe una región abierta no vacía $G_D \subset A_D$ tal que, si $(x, y) \in G_D$, entonces el primer paso del modelo SR devuelve un valor $\tilde{\alpha} < \beta$.*

Según el teorema, cualquier agente que parta de la región G_D será dirigido inicialmente contra el muro w . El modelo SR es entonces subóptimo; es decir, su capacidad cognitiva está por debajo de muchos seres vivos y, por lo tanto, el algoritmo no es capaz de describir el comportamiento humano, al menos en ciertas situaciones relativamente simples.

Para la demostración del teorema se procederá de manera segmentada, probando por partes una serie de lemas y proposiciones cuyas afirmaciones conducirán a la prueba del resultado global enunciado.

4.2.2. Posibles direcciones iniciales del agente

Una primera aproximación consiste en discernir los posibles valores $\tilde{\alpha}$ que el modelo SR puede proporcionar como dirección óptima del agente a la hora de calcular el primer paso que éste ha de dar:

Lema 4.1 (Posibles valores de $\tilde{\alpha}$). *Bajo las condiciones del Escenario 4.2, el modelo SR dirigirá inicialmente al agente situado en $(x, y) \in A_D$ hacia una dirección $\tilde{\alpha}$ tal que*

$$\tilde{\alpha} \in \left\{ \alpha_0, \arctan \left(-\frac{D}{y} \operatorname{sen}(\alpha_0) \right), \beta \right\}.$$

Demostración. Fijada una ubicación inicial del agente $(x, y) \in A_D$, las funciones $f : [\phi, \phi] \rightarrow [0, D]$ y $d : [\phi, \phi] \rightarrow [0, +\infty)$ (usadas por el algoritmo para la determinación de la dirección del primer paso) quedan, para esta situación particular:

$$f(\alpha) = \begin{cases} -\frac{y}{\cos \alpha} & \text{si } \alpha \in (\gamma, \beta) \\ D & \text{si } \alpha \in [-\phi, \gamma] \cup [\beta, \phi] \end{cases}$$

$$d(\alpha) = \begin{cases} D^2 + \frac{y^2}{\cos^2 \alpha} + 2yD \frac{\cos(\alpha - \alpha_0)}{\cos \alpha} & \text{si } \alpha \in (\gamma, \beta) \\ 2D^2 (1 - \cos(\alpha - \alpha_0)) & \text{si } \alpha \in [-\phi, \gamma] \cup [\beta, \phi] \end{cases}$$

Tanto f como d son funciones continuas en $[-\phi, \phi] - \{\beta\}$ no derivables en γ . La elección de $\tilde{\alpha}$ se lleva a cabo encontrando el mínimo global de la función d en $[-\phi, \phi]$. Este mínimo global podrá alcanzarse bien en alguno

de sus mínimos locales, bien en alguno de los puntos extremos $-\phi, \gamma, \beta, \phi$, o incluso podría no alcanzarse, dada la discontinuidad de d en β . Para que esto no ocurra, redefiniremos el valor de d en β como

$$d(\beta) := \min\{d(\beta^-), d(\beta^+)\},$$

siendo

$$d(\beta^-) = D^2 + x^2 + y^2 + 2D(x \sin \alpha_0 + y \cos \alpha_0),$$

$$d(\beta^+) = 2D^2 \left(1 + \frac{1}{\|(x, y)\|} (x \sin \alpha_0 + y \cos \alpha_0) \right).$$

Comencemos analizando los posibles mínimos locales de la función d , a partir de la expresión analítica de su derivada:

$$d'(\alpha) = \begin{cases} \frac{2y}{\cos^2 \alpha} (y \tan \alpha + D \sin \alpha_0) & \text{si } \alpha \in (\gamma, \beta) \\ 2D^2 \sin(\alpha - \alpha_0) & \text{si } \alpha \in [-\phi, \gamma] \cup [\beta, \phi] \end{cases}$$

- $m_1 = \alpha_0$ es mínimo local de d si $m_1 < \gamma$ o si $\beta < m_1$. En caso de serlo,

$$d(m_1) = 0.$$

- $m_2 = \arctan\left(-\frac{D}{y} \sin \alpha_0\right)$ es mínimo local de d si $\gamma < m_2 < \beta$. En caso de serlo,

$$d(m_2) = (D \cos \alpha_0 + y)^2.$$

Obsérvese que si m_1 es mínimo local entonces también es mínimo global, pues d tomaría en m_1 el mínimo valor alcanzable por la función.

En cuanto a la posibilidad de que el mínimo global de d se alcance en uno de los extremos, por un lado, teniendo en cuenta que $d|_{[-\phi, \gamma] \cup [\beta, \phi]}$ es función convexa, $-\phi$ y ϕ quedan descartados. Por otro lado, supongamos que γ fuera mínimo global; entonces, por ser d continua en γ , tendrían que darse simultáneamente las condiciones $d'(\gamma^-) < 0$ y $d'(\gamma^+) > 0$. Pero

$$d'(\gamma^+) = -2 \frac{D^3}{|y|} (\sin \alpha_0 - \sin \gamma) > 0 \iff \sin \alpha_0 < \sin \gamma \iff \alpha_0 < \gamma.$$

Luego si $d'(\gamma^+) > 0$, entonces $d|_{[-\phi, \gamma]}$, que es función convexa, alcanza un mínimo local (luego global) en $m_1 = \alpha_0 < \gamma$. Por tanto, $d'(\gamma^-) > 0$ y en consecuencia γ también queda descartado como candidato a mínimo global de d . Queda entonces un único candidato por añadir a la lista de posibles mínimos globales:

- $m_3 = \beta$ es mínimo global cuando d no tiene mínimos locales o bien cuando, siendo m_2 su único mínimo local, $d(m_3) = \min\{d(\beta^-), d(\beta^+)\} \leq d(m_2)$. Si esto ocurriese,

$$d(m_3) = d(\beta^+) = 2D^2 \left(1 + \frac{1}{\|(x, y)\|} (x \sin \alpha_0 + y \cos \alpha_0) \right).$$

□

4.2.3. Discusión de los mínimos locales de la función d

Una vez acotado el conjunto de posibles valores de $\tilde{\alpha}$, interesa conocer para qué casos el algoritmo se comporta de manera razonable, esto es, devolviendo una dirección inicial $\tilde{\alpha}$ para dar el primer paso que verifique $\tilde{\alpha} \geq \beta$. Como se ha comentado en la demostración del Lema 4.1, la función d asociada podría tener uno o dos mínimos locales, o incluso podría carecer de ellos. Los dos siguientes lemas tendrán como propósito delimitar aquellas regiones de partida que induzcan uno u otro caso.

Lema 4.2 (Región donde m_1 es mínimo local de d). *En las condiciones dadas para el Escenario 4.2, $m_1 = \alpha_0$ será mínimo local de d cuando el agente parta de una posición inicial (x, y) que **verifique una de las siguientes condiciones**:*

- $x > x_T + \left(1 - \frac{y_T}{y}\right) \sqrt{D^2 - y^2}$
- $y > \frac{y_T}{x_T} x$

Demostración. Como se ha comentado en la demostración del Lema 4.1, m_1 es mínimo local si $\alpha_0 < \gamma$ o si $\beta < \alpha_0$:

- $\alpha_0 < \gamma \iff \arctan\left(\frac{x - x_T}{y - y_T}\right) < \arctan\left(\frac{1}{y} \sqrt{D^2 - y^2}\right) \iff$
 $\iff x > x_T + \left(1 - \frac{y_T}{y}\right) \sqrt{D^2 - y^2}$
- $\beta < \alpha_0 \iff \arctan\left(\frac{x}{y}\right) < \arctan\left(\frac{x_T - x}{y_T - y}\right) \iff y > \frac{y_T}{x_T} x$

Ambos casos han sido razonados teniendo en cuenta que la función \arctan es creciente en todo su dominio.

□

Lema 4.3 (Región donde m_2 es mínimo local de d). *En las condiciones dadas para el Escenario 4.2, $m_2 = \arctan\left(-\frac{D}{y}\sin(\alpha_0)\right)$ será mínimo local de d cuando el agente parta de una posición inicial (x, y) que **verifique conjuntamente las condiciones siguientes**:*

- $x < x_T + \left(1 - \frac{y_T}{y}\right) \sqrt{D^2 - y^2}$
- Si $x < x_T$, $y < y_T + \left(\frac{x_T}{x} - 1\right) \sqrt{D^2 - x^2}$
- Si $x > 0$, $y > y_T + \left(\frac{x_T}{x} - 1\right) \sqrt{D^2 - x^2}$

Demostración. Como se ha comentado en la demostración del Lema 4.1, m_2 es mínimo local si $\gamma < m_2 < \beta$:

- $\gamma = \arctan\left(-\frac{1}{|y|}\sqrt{D^2 - y^2}\right) < \arctan\left(\frac{D}{|y|}\sin(\alpha_0)\right) \iff$
 $\iff -\sqrt{D^2 - y^2} < D \sin(\alpha_0)$.

Llegados a este punto, debemos distinguir dos casos:

- Si $x < x_T$, la última desigualdad se verifica trivialmente, pues el término de la izquierda es negativo y el de la derecha es positivo.
- Si $x \geq x_T$, los dos términos de la última desigualdad son negativos; ésta se verificará entonces si y sólo si:

$$\begin{aligned} D^2 - y^2 > D^2 \sin^2(\alpha_0) &\iff D^2 \cos^2(\alpha_0) > y^2 \iff \\ \iff D \cos(\alpha_0) > -y &\iff -\alpha_0 < \arccos\left(-\frac{y}{D}\right) \iff \alpha_0 > \gamma. \end{aligned}$$

Ahora, de manera análoga al primer punto de la demostración del Lema 4.2, esto es equivalente a:

$$x < x_T + \left(1 - \frac{y_T}{y}\right) \sqrt{D^2 - y^2}.$$

- $\arctan\left(-\frac{D}{y}\sin(\alpha_0)\right) < \arctan\left(\frac{x}{y}\right) = \beta \iff x < -D \sin(\alpha_0)$.

Llegados a este punto, debemos distinguir tres casos:

- Si $x < x_T$, los dos términos de la última desigualdad son negativos, luego la desigualdad se verificará si y sólo si:

$$\begin{aligned} x^2 > D^2 \sin^2(\alpha_0) &\iff x^2 > D^2 \frac{(x_T - x)^2}{(x_T - x)^2 + (y_T - y)^2} \iff \\ \iff y < y_T + \left(\frac{x_T}{x} - 1\right) \sqrt{D^2 - x^2}. \end{aligned}$$

- Si $x_T \leq x \leq 0$, la última desigualdad se verifica de manera trivial, bien por ser el término de la izquierda estrictamente negativo y el de la derecha nulo, bien por ser el término de la izquierda negativo y el de la derecha positivo, bien por ser el de la izquierda nulo y el de la derecha estrictamente positivo.
- Si $x > 0$, ambos términos son positivos, luego la desigualdad se verificará si y sólo si:

$$x^2 < D^2 \sen^2(\alpha_0) \iff y > y_T + \left(\frac{x_T}{x} - 1\right) \sqrt{D^2 - x^2}.$$

En ambos casos nos hemos apoyado en el hecho de que la función arctan es estrictamente creciente en todo su dominio; para el primer caso, además, hemos utilizado que la función arc cos es estrictamente decreciente si se considera su imagen dentro del rango no negativo $[0^\circ, 180^\circ]$. \square

A partir de los resultados obtenidos en los Lemas 4.2 y 4.3 es posible realizar una partición de la región de alcance A_D , distinguiendo aquellas posiciones iniciales del agente que induzcan en el primer paso del algoritmo una función d con dos, uno o ningún mínimo local.

Notación 4.1. *En las condiciones enunciadas en el Escenario 4.2, fijado $D > 0$, para cada $k \in \{1, 2\}$ denotaremos*

$$\Omega_k = \{(x, y) \in A_D : m_k \text{ es mínimo local de } d \text{ si el agente parte de } (x, y)\},$$

$$\text{siendo } m_1 = \alpha_0, m_2 = \arctan\left(-\frac{D}{y} \sen(\alpha_0)\right).$$

Los Lemas 4.2 y 4.3 establecen las ecuaciones de Ω_1 y Ω_2 . La Figura 4.3 muestra una representación visual de estos conjuntos para el caso particular $D = 10$, $(x_T, y_T) = (-2, 1)$.

Notación 4.2. *Bajo los supuestos considerados en el Escenario 4.2, fijado $D > 0$, para cada par $(x, y) \in \mathbb{R}^2$ denotaremos:*

$$\delta_0 := \delta_0(x, y) = \|(x, y)\|,$$

$$\delta_T := \delta_T(x, y) = \|(x - x_T, y - y_T)\|.$$

Obsérvese que δ_0 y δ_T representan, respectivamente, las distancias al origen y al target. También se usará, de aquí en adelante, la reducción:

$$\mu_D(x, y) := 2\delta_T[x(x - x_T) + y(y - y_T)] + \delta_0 \left(\left[\left(\frac{\delta_T}{D} - 1 \right) y + y_T \right]^2 - 2\delta_T^2 \right).$$

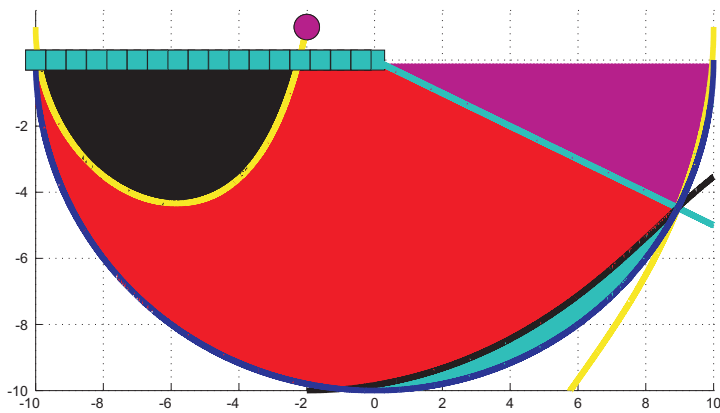


Figura 4.3: Discriminación de las zonas de A_D en las que existen dos, uno o ningún mínimo local para la función d , fijados $D = 10$ y $(x_T, y_T) = (-2, 1)$. En bloques de color cian, la representación del obstáculo extendido W . El círculo magenta indica la posición del *target*. La curva de color azul oscuro delimita la región de alcance A_D . La curva amarilla tiene por ecuación $y = y_T + \left(\frac{x_T}{x} - 1\right) \sqrt{D^2 - x^2}$, la negra, $x = x_T + \left(1 - \frac{y_T}{y}\right) \sqrt{D^2 - y^2}$ y la recta cian, $y = \frac{y_T}{x_T} x$. Si el agente parte inicialmente de la región coloreada de negro, la función d definida en el primer paso del algoritmo de resolución del modelo SR carece de mínimos locales; si parte de la región coloreada de cian, d tendrá un único mínimo local en m_1 ; si parte de la región coloreada de rojo, d tendrá un único mínimo local en m_2 ; si parte de la región coloreada de magenta, d tendrá dos mínimos locales: uno en m_1 y otro en m_2 .

4.2.4. Discriminación de las zonas de fallo y acierto

En base a todo lo estudiado hasta ahora y de acuerdo a las notaciones establecidas, se puede concluir el siguiente resultado:

Proposición 4.1 (Valor de $\tilde{\alpha}$ en función de la posición inicial del agente). *Bajo las condiciones dadas en el Escenario 4.2, la primera dirección $\tilde{\alpha}$ en la que el modelo SR dirigirá al agente si su ubicación inicial es (x, y) será:*

$$\tilde{\alpha} := \tilde{\alpha}(x, y) = \begin{cases} \alpha_0 & \text{si } (x, y) \in \Omega_1 \\ \arctan\left(-\frac{y}{D} \sin(\alpha_0)\right) & \text{si } (x, y) \in \Omega_2 - \Omega_1 \text{ y } \mu_D(x, y) < 0. \\ \beta & \text{en otro caso} \end{cases}$$

*En consecuencia, el algoritmo fallará en su primer paso si y sólo si el agente ocupa una posición inicial $(x, y) \in A_D$ que **verifica una de las siguientes condiciones:***

- $(x, y) \in \Omega_1 \wedge y < \frac{y_T}{x_T} x$
- $(x, y) \in \Omega_2 - \Omega_1 \wedge \mu_D(x, y) < 0$

Demostración. La dirección inicial del agente será aquélla que haga mínimo el valor de la función d definida para el primer paso del algoritmo de resolución del modelo SR.

Por un lado, como ya se comentó en la demostración del Lema 4.1, el hecho de que d tenga un mínimo local en m_1 implica que d alcance también aquí su mínimo global, puesto que $d(m_1) = 0$. Luego cuando el agente parta inicialmente de cualquier punto de la región Ω_1 , el mínimo global de la función d definida para el primer paso del algoritmo de resolución del modelo SR se alcanzaría en $m_1 = \alpha_0$.

Por otra parte, si restringimos el estudio a la región $\Omega_2 - \Omega_1$ caben dos posibilidades: que el mínimo global se alcance en el mínimo local m_2 (obsérvese que aquí $m_2 < \beta$) o que, pese a la existencia de este mínimo local, el salto de d en β sea tal que $d(\beta) < d(m_2)$:

$$\begin{aligned} d(m_2) < d(\beta) &\iff (D \cos \alpha_0 + y)^2 < 2D^2 \left(1 + \frac{1}{\delta_0} (x \sin \alpha_0 + y \cos \alpha_0)\right) \iff \\ &\iff 2D^2 (x \sin \alpha_0 + y \cos \alpha_0) > \delta_0 [(D \cos \alpha_0 + y)^2 - 2D^2] \iff \\ &\iff -\frac{2D^2}{\delta_T} [x(x - x_T) + y(y - y_T)] > \delta_0 \left[\left(y - \frac{D(y - y_T)}{\delta_T}\right)^2 - 2D^2 \right] \iff \\ &\iff 2D^2 \delta_T [x(x - x_T) + y(y - y_T)] + \delta_0 \left[(\delta_T - D)y + Dy_T \right]^2 - 2D^2 \delta_T^2 < 0 \end{aligned}$$

Se tiene entonces que el mínimo global de d será $m_2 = \arctan\left(-\frac{y}{D} \operatorname{sen}(\alpha_0)\right)$ si $\mu_D(x, y) < 0$ y β en caso contrario.

Queda por último analizar la región $A_D - (\Omega_1 \cup \Omega_2)$. En ella no existen mínimos locales, luego apoyándonos en los tres lemas previos, en caso de que el agente parta desde ella, el mínimo global de d se alcanzará necesariamente en β .

Habiendo obtenido la expresión de la dirección inicial $\tilde{\alpha}$ en función de la posición inicial del agente, estamos en condiciones de discriminar aquellas regiones para las que el modelo va a ser capaz de proporcionar una solución inicial razonable de aquéllas en las que no.

Atendiendo al primer posible valor de $\tilde{\alpha}$, surge ya una primera región de fallo: aquella en la que el mínimo global de d es α_0 , pero $\alpha_0 < \beta$ (en nuestro ejemplo concreto, la región cian de la Figura 4.3 situada por debajo de la recta amarilla). No obstante, este fallo es justificable atendiendo a la distancia del horizonte D fijada. Debido a dicha limitación, en esta región el agente no es capaz de alcanzar a ver ningún obstáculo que obstruya la trayectoria recta que lo conduciría a su destino, por lo que es justificable que dirija su primer paso en esa dirección (i.e. $\tilde{\alpha} = \alpha_0 < \beta$).

Además, el algoritmo fallará (ahora sí, de manera no justificada) siempre que el agente parta de una posición inicial $(x, y) \in \Omega_2 - \Omega_1$ que verifique la inecuación $\mu_D(x, y) < 0$: en este caso $\tilde{\alpha} = m_2$ y, por encontrarse el agente en Ω_2 , $m_2 < \beta$.

□

Volviendo a nuestro ejemplo particular, la Figura 4.3 es insuficiente para discriminar esta nueva región de fallo. Por ello, en la Figura 4.4 se ha añadido la curva separatriz de ecuación implícita $\mu_D(x, y) = 0$, de modo que se establezca una partición sobre la zona coloreada de rojo y así se puedan distinguir de manera visual aquellas zonas para las que el algoritmo SR va a saber encontrar una solución razonable de aquéllas para las que no. Para saber cuál de las regiones separadas por esta curva corresponde con un comportamiento ($\mu_D(x, y) > 0$) u otro ($\mu_D(x, y) < 0$) basta observar que para cualquier valor de D y para cualquier posición (x_T, y_T) del *target*, $(x_T, 0) \in \partial A_D$ y

$$\mu_D(x_T, 0) = 0 - x_T (y_T^2 - 2y_T^2) = x_T y_T^2 < 0,$$

luego ambas zonas quedan unívocamente determinadas.

Obsérvese que la Figura 4.4 explica el comportamiento del agente al que en el Escenario 4.1 se le había implementado el algoritmo SR (véase la Figura 4.1). Otras disposiciones recogidas en el Escenario General 4.2 pueden observarse en la Figura 4.5.

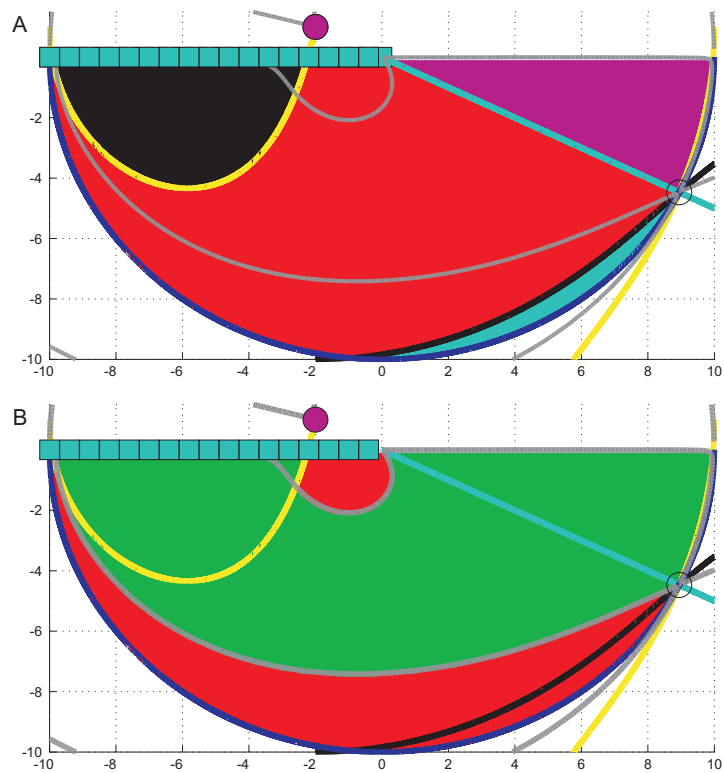


Figura 4.4: (A) Se ha añadido a la Figura 4.3, en gris, la curva de ecuación implícita $\mu_D(x, y) = 0$. (B) Separación de las zonas en las que el algoritmo falla o acierta en su primer paso: cuando el agente parte de cualquier punto coloreado de verde, el algoritmo se comporta inicialmente de manera razonable; cuando parte de una posición roja, el algoritmo lo lanza de inicio contra el muro, en vez de dirigirlo hacia la zona abierta. Para la representación de las curvas delimitantes se ha empleado el mismo código de color que en la Figura 4.3.

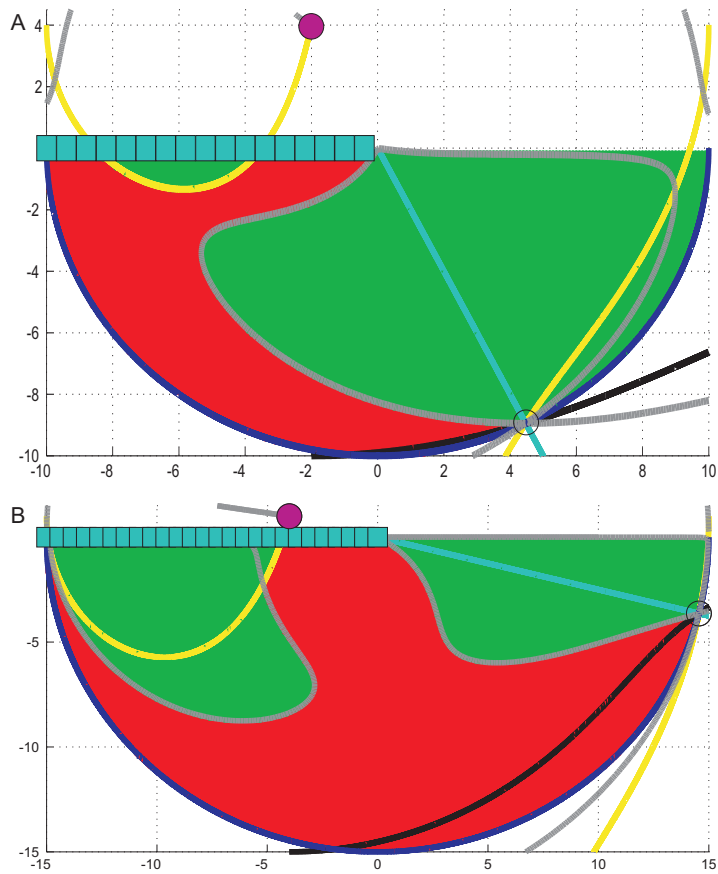


Figura 4.5: En verde, las posiciones iniciales del agente que son resueltas de manera razonable por el primer paso del modelo SR; en rojo, las posiciones iniciales para las que el modelo sugiere dirigir al agente contra el muro w . Para las curvas que delimitan estas regiones se ha mantenido el mismo código de color que se usó previamente en las Figuras 4.3 y 4.4. (A) $D = 10$, $(x_T, y_T) = (-2, 4)$. (B) $D = 15$, $(x_T, y_T) = (-4, 1)$.

4.2.5. Demostración del resultado principal

Con todo lo visto hasta ahora, demostrar el Teorema 4.1 se reduce a la siguiente prueba:

Demostración del Teorema 4.1. Por un lado, todo punto en A_D de abscisa x_T pertenece a la región $\Omega_2 - \Omega_1$, por no verificar ninguna de las condiciones impuestas en el Lema 4.2 pero sí todas las impuestas en el Lema 4.3:

$$y < 0 < y_T \implies -\frac{y_T}{y} > 0 \implies x_T < x_T + \left(1 - \frac{y_T}{y}\right) \sqrt{D^2 - y^2}$$

$$y < y_T \implies y < \frac{y_T}{x_T} x_T$$

Por otro lado, como ya se ha comentado anteriormente, $(x_T, 0)$ es punto del abierto $\{(x, y) \in \mathbb{R}^2 : \mu_D(x, y) < 0\} \subset \mathbb{R}^2$, pero además $(x_T, 0) \in \partial A_D$. Entonces, sean cuales sean la disposición del *target* (x_T, y_T) y la “distancia de horizonte” D del agente, siempre existirá un entorno abierto $B_\epsilon(x_T, 0) \subset \{(x, y) \in \mathbb{R}^2 : \mu_D(x, y) < 0\}$ tal que $G_D := B_\epsilon(x_T, 0) \cap A_D \neq \emptyset$.

En otras palabras, siempre va a existir una región, por pequeña que sea, para la cual el algoritmo se comporte de manera absurda a la hora de elegir el primer paso que debe dar el agente para alcanzar su destino. □

4.3. Limitaciones no dependientes de la distancia del horizonte fijada

Obsérvese que en los ejemplos analizados siempre surge una región de fallo. Podría pensarse que aumentando el parámetro libre $D > 0$ esta región tendería a desaparecer. Sin embargo puede mostrarse que esto no es así.

4.3.1. Resultado principal

El último propósito de este apartado será demostrar la existencia de zonas desde las cuales cualquier agente en el que se haya implementado el modelo SR va a dirigir inicialmente sus pasos hacia el muro w , independientemente de su alcance de visión, esto es, del valor de D .

Teorema 4.2. *Considérese el Escenario 4.2 con posibilidad de aumentar libremente la distancia del horizonte del agente implicado a cualquier valor $D' > D$. Existe una región abierta $F \subset A_D$, independiente del valor de D' , tal que, si $(x, y) \in F$, entonces el primer paso del modelo SR devuelve un valor $\tilde{\alpha} < \beta$.*

4.3.2. Construcción analítica de una subregión de fallo dependiente de la distancia del horizonte

Como en el subapartado anterior, procederemos de manera segmentada, probando una serie de resultados previos que conducirán a la demostración de este teorema.

Notación 4.3. Sean $D > 0$, $(x_T, y_T) \in (-D, 0) \times (0, D)$. Para cada $(x, y) \in [(-D, D) - \{0\}] \times \mathbb{R}$ denotaremos:

$$\lambda_D(x, y) := y - y_T + \left(1 - \frac{x_T}{x}\right) \sqrt{D^2 - x^2}.$$

De manera análoga, para cada $(x, y) \in \mathbb{R} \times [(-D, D) - \{0\}]$ denotaremos:

$$\xi_D(x, y) := x - x_T + \left(\frac{y_T}{y} - 1\right) \sqrt{D^2 - y^2}.$$

Por último, para cada $(x, y) \in \mathbb{R}^2$ denotaremos:

$$\omega(x, y) := y_T x - x_T y.$$

Las curvas $\{(x, y) \in \mathbb{R}^2 : \lambda_D(x, y) = 0\}$, $\{(x, y) \in \mathbb{R}^2 : \xi_D(x, y) = 0\}$ y $\{(x, y) \in \mathbb{R}^2 : \omega(x, y) = 0\}$ han sido representadas con líneas amarilla, negra y cian, respectivamente, en las Figuras 4.3, 4.4 y 4.5.

Lema 4.4. Sean $D > 0$, $(x_T, y_T) \in (-D, 0) \times (0, D)$. Si $D' \geq D$ y $(x, y) \in \mathbb{R}^2$ es punto de la curva $\{\lambda_{D'}(x, y) = 0\}$, entonces:

- $x_T \leq x < 0 \implies y \geq y_T$
- $0 < x < \frac{D|x_T|}{\|(x_T, y_T)\|} \implies y \leq -\sqrt{D^2 - x^2}$

Demostración.

$$x_T \leq x < 0 \implies \frac{x_T}{x} - 1 \geq 0 \implies y = y_T + \left(\frac{x_T}{x} - 1\right) \sqrt{D'^2 - x^2} \geq y_T.$$

$$0 < x < \frac{D|x_T|}{\|(x_T, y_T)\|} \implies \frac{1}{x^2} > \frac{x_T^2 + y_T^2}{D^2 x_T^2} \implies \sqrt{\frac{D^2}{x^2} - 1} > \sqrt{\frac{x_T^2 + y_T^2}{x_T^2} - 1} \implies$$

$$\implies |x_T| \sqrt{\frac{D^2}{x^2} - 1} > y_T \implies y_T + \frac{x_T}{x} \sqrt{D^2 - x^2} < 0 \implies$$

$$\implies y_T + \left(\frac{x_T}{x} - 1\right) \sqrt{D^2 - x^2} \leq -\sqrt{D^2 - x^2} \implies$$

$$\implies y = y_T + \left(\frac{x_T}{x} - 1\right) \sqrt{D'^2 - x^2} \leq -\sqrt{D^2 - x^2}.$$

□

Lema 4.5. Sean $D > 0$, $(x_T, y_T) \in (-D, 0) \times (0, D)$. Si $D' \geq D$ y $(x, y) \in \mathbb{R}^2$ es punto de la curva $\{\xi_{D'}(x, y) = 0\}$, entonces

$$y < 0 \implies x_T + \sqrt{D^2 - (y - y_T)^2} < x.$$

Demostración.

$$\begin{aligned} 0 < -y < y_T - y &\implies y^2 D^2 < (y_T - y)^2 D^2 \implies \\ &\implies y^2 D^2 - y^2 (y_T - y)^2 < (y_T - y)^2 D^2 - y^2 (y_T - y)^2 \implies \\ &\implies |y| \sqrt{D^2 - (y_T - y)^2} < (y_T - y) \sqrt{D^2 - y^2} \implies \\ &\implies x_T + \sqrt{D^2 - (y_T - y)^2} < x_T + \left(1 - \frac{y_T}{y}\right) \sqrt{D^2 - y^2} \implies \\ &\implies x_T + \sqrt{D^2 - (y_T - y)^2} < x = x_T + \left(1 - \frac{y_T}{y}\right) \sqrt{D'^2 - y^2}. \end{aligned}$$

□

Definición 4.2. Bajo los supuestos establecidos en el Escenario 4.2, definimos la región correspondiente a aquellos puntos de A_D cuya distancia al target es inferior a D :

$$C_D := \{(x, y) \in A_D : \delta_T < D\} \subset A_D.$$

Dentro de ella se define, para cada $D' \geq D$, la subregión $F_{D'} \subset C_D$ como aquella delimitada por las curvas $\{\mu_{D'}(x, y) = 0\}$, $\{\lambda_{D'}(x, y) = 0 \wedge x < x_T\}$ y $\{\omega(x, y) = 0\}$:

$$F_{D'} = \{(x, y) \in C_D : \mu_{D'}(x, y) < 0 \wedge (\lambda_{D'}(x, y) < 0 \vee x \geq x_T) \wedge \omega(x, y) < 0\}.$$

Asimismo, definiremos el conjunto-límite $F \subset C_D$:

$$F := \lim_{D' \rightarrow +\infty} F_{D'}.$$

Proposición 4.2. Considérese la situación descrita en el Escenario 4.2. Si el agente implicado se sitúa inicialmente en $(x, y) \in C_D \subset A_D$ y se amplía su distancia del horizonte a $D' \geq D$, el algoritmo del modelo SR fallará inicialmente si y sólo si

$$(x, y) \in F_{D'}.$$

Demostración. De lo estudiado en la Proposición 4.1 se desprende que, para que el algoritmo falle en su primer paso, necesariamente el agente ha de situarse inicialmente en una posición (x, y) tal que $\omega(x, y) < 0$: si no hay obstáculo alguno que se interponga en la dirección al *target*, el algoritmo sabrá escoger como óptima la trayectoria recta hasta el objetivo.

Por otro lado, teniendo en cuenta que $\{\omega(x, y) = 0\}$ corta a la frontera de A_D en $(0, 0)$ y $\left(-\frac{Dx_T}{\|(x_T, y_T)\|}, -\frac{Dy_T}{\|(x_T, y_T)\|}\right)$, la conclusión geométrica que se puede extraer del Lema 4.4 es que la curva $\{\lambda_{D'}(x, y) = 0\}$ es exterior a $A_D \cap \{\omega(x, y) < 0\}$ cuando $x \geq x_T$: si $x_T < x < 0$, todos los puntos de A_D quedan por debajo de ella, mientras que si $0 < x$, todos los puntos de $A_D \cap \{\omega(x, y) < 0\}$ quedan por encima.

Finalmente, del Lema 4.5 puede concluirse que los puntos de la curva $\{\xi_{D'}(x, y) = 0\}$ son exteriores a la circunferencia de centro (x_T, y_T) y radio D cuando $y < 0$; en definitiva, esta curva queda fuera de la región a la que hemos restringido el estudio, C_D .

Así pues, de las dos alternativas que la Proposición 4.1 establecía para describir las zonas de fallo del algoritmo SR aplicado al Escenario 4.2 (con el valor D aumentado a D'), la primera de ellas no tiene cabida si nos restringimos a la subregión C_D , y la segunda de ellas puede expresarse analíticamente de manera más sencilla, evitando redundancias propias de C_D : en esta subregión, para que (x, y) sea punto de la región $\Omega_2 - \Omega_1$ basta con que se verifique bien $x \geq x_T$, bien $\lambda_{D'}(x, y) < 0$. Queda de este modo probado que, de las dos condiciones de fallo excluyentes definidas en la Proposición 4.1, el conjunto de puntos que verifican en C_D la primera de ellas es vacío, mientras que el conjunto de puntos que verifican la segunda es exactamente $F_{D'}$. \square

4.3.3. Evolución de la subregión de fallo a medida que aumenta la distancia del horizonte

Estudiaremos cómo varía la familia de conjuntos $\{F_{D'}\}_{D' \geq D}$ a medida que aumenta el parámetro libre D' . Como puede recogerse del resultado expuesto en la Proposición 4.2, $F_{D'}$ corresponde a una subregión de fallo del algoritmo cuando se fija D' como distancia del horizonte del agente implicado. El conjunto-límite F representará entonces ciertas zonas en las que el algoritmo continuaría fallando por mucho que se aumente este parámetro.

Lema 4.6. *Bajo las condiciones establecidas en el Escenario 4.2, fijado cualquier par $(x, y) \in C_D$, defínase la función escalar $\mu : (D, +\infty) \rightarrow \mathbb{R}$ como aquella que asigna a cada valor $s > D$ la evaluación de μ_s en (x, y) , i.e. $\mu(s) := \mu_s(x, y)$. Análogamente, y obviando los casos en los que $x \geq x_T$, defínase la función escalar $\lambda : (D, +\infty) \rightarrow \mathbb{R}$ como $\lambda(s) := \lambda_s(x, y)$. Se verifica que μ y λ son funciones crecientes en todo su dominio.*

Demostración. Para probar que la función escalar μ es creciente, teniendo en cuenta que, por pertenecer (x, y) a C_D , $\delta_T < s \in (D, +\infty)$:

$$\frac{d\mu}{ds} = 2\delta_0\delta_T \frac{y^2}{s^2} \left(\frac{y_T}{|y|} + 1 - \frac{\delta_T}{s} \right) > 0, \forall s \in (D, +\infty).$$

Análogamente:

$$\frac{d\lambda}{ds} = \left(1 - \frac{x_T}{x} \right) \frac{s}{\sqrt{s^2 - x^2}} > 0, \forall s \in (D, +\infty), \text{ si } x < x_T.$$

□

Lema 4.7. *Bajo las condiciones descritas en el Escenario 4.2, para cualesquiera $D_1, D_2 \in (D, +\infty)$:*

$$D_1 < D_2 \implies F_{D_2} \subset F_{D_1}.$$

Demostración. Tómense dos valores cualesquiera $D_1, D_2 \in (D, +\infty)$, $D_1 < D_2$, y distínganse dos familias de casos, que comprenden la totalidad de ellos; en ambas se aplicará el resultado obtenido en el Lema 4.6:

- Si $(x, y) \in C_D \cap \{x < x_T\}$:

$$(x, y) \in F_{D_2} \implies \mu_{D_2}(x, y) < 0 \text{ y } \lambda_{D_2}(x, y) < 0 \implies$$

$$\mu_{D_1}(x, y) < 0 \text{ y } \lambda_{D_1}(x, y) < 0 \implies (x, y) \in F_{D_1}$$

- Si $(x, y) \in C_D \cap \{x \geq x_T\}$:

$$(x, y) \in F_{D_2} \implies \mu_{D_2}(x, y) < 0 \implies$$

$$\mu_{D_1}(x, y) < 0 \implies (x, y) \in F_{D_1}.$$

□

4.3.4. Demostración del resultado principal

Interesa, al menos, verificar que $F \neq \emptyset$. El Teorema 4.1 probó que para cualquier situación recogida en el Escenario General 4.2 existe siempre posibilidad de disponer inicialmente al agente de manera que el modelo SR decida dirigirlo de inicio contra el obstáculo w . Ahora bien, muchos de estos fallos del modelo podrían evitarse sin más que ampliar la distancia del horizonte D del agente implicado; sin embargo, la prueba del Teorema 4.2 supondrá que otros tantos fallos seguirán dándose por mucho que se aumente este parámetro:

Demostración del Teorema 4.2. Tal y como se ha construido el conjunto-límite F introducido la Definición 4.2, una consecuencia inmediata que se puede extraer del Lema 4.7 es que $\{F_{D'}\}_{D' \geq D}$ es una familia de subconjuntos anidados, $F = \bigcap_{D' \geq D} F_{D'}$ y, por tanto, se verifica

$$F \subset F_{D'}, \forall D' \geq D.$$

El conjunto-límite F verificará entonces la implicación impuesta en el enunciado de este teorema. Queda únicamente pendiente la demostración de la existencia de puntos contenidos en F . En concreto, probaremos que el segmento no vacío

$$\left(\{x_T\} \times \left[\frac{1}{\sqrt{3}}x_T, 0 \right] \right) \cap C_D$$

está contenido en F .

Comenzaremos estudiando las curvas-límite que delimitan F , al que podemos referirnos, de acuerdo a las conclusiones del Lema 4.7, como *región de fallo común a cualquier valor del parámetro D* . Por un lado:

$$\lambda_D(x, y) = 0 \iff y - y_T = -(x - x_T) \frac{\sqrt{D^2 - x^2}}{x} \iff x = x_T - x \frac{y - y_T}{\sqrt{D^2 - x^2}}.$$

Haciendo $D \rightarrow +\infty$, esta primera curva tenderá a la asíntota vertical de ecuación

$$x = x_T.$$

Del mismo modo, nos interesará también analizar el límite de la curva $\{\mu_D(x, y) = 0\}$. Denotando

$$\bar{\mu}(x, y) := \lim_{D \rightarrow +\infty} \mu_D(x, y) = 2\delta_T[x(x - x_T) + y(y - y_T)] + \delta_0 \left([y - y_T]^2 - 2\delta_T^2 \right),$$

podemos expresar analíticamente F como:

$$F = \{(x, y) \in C_D : x \geq x_T \wedge \bar{\mu}(x, y) \leq 0\}.$$

Nos proponemos en primer lugar encontrar el corte de la curva-límite $\{\bar{\mu}(x, y) = 0\}$ con la asíntota $\{x = x_T\}$:

$$\begin{aligned} \bar{\mu}(x_T, y) &= 2(y_T - y)(y(y - y_T)) + \delta_0 ([y_T - y]^2 - 2(y_T - y)^2) = \\ &= -(2y + \delta_0)(y_T - y)^2 = 0 \iff y = y_T \vee -2y = \delta_0. \end{aligned}$$

La primera de las soluciones corresponde al *target*, que es punto común a toda la familia de curvas $\{\mu_D(x, y) = 0\}$, luego también lo es de la curva límite $\{\bar{\mu}(x, y) = 0\}$; pero (x_T, y_T) queda fuera de la región de estudio. La otra solución, sin embargo, sí puede pertenecer a C_D y, por tanto, ser

un punto de la adherencia de F (que es un cerrado de la topología usual inducida en C_D):

$$-2y = \delta_0 \iff y = \frac{1}{\sqrt{3}}x_T < 0$$

Luego concluimos que el corte de las dos curvas-límite se da en $\left(x_T, \frac{1}{\sqrt{3}}x_T\right)$.

Ahora bien, dada la continuidad de $\bar{\mu}(x_T, y)$ respecto de su segunda variable, podemos afirmar que $\bar{\mu}(x_T, y)$ no cambia de signo cuando y toma valores en el intervalo $\left(\frac{1}{\sqrt{3}}x_T, y_T\right) \ni 0$. Y como $\bar{\mu}(x_T, 0) = x_T y_T^2 < 0$, en efecto,

$$\emptyset \neq \left(\{x_T\} \times \left[\frac{1}{\sqrt{3}}x_T, 0\right]\right) \cap C_D \subset F,$$

tal y como se quería demostrar. □

En definitiva: para cualquiera que sea la situación del *target* (x_T, y_T) , si el agente se sitúa inicialmente en un punto de abscisa x_T y ordenada $y \in \left[\frac{1}{\sqrt{3}}x_T, 0\right)$, $y > \max\{-\sqrt{D^2 - x_T^2}, y_T - D\}$, el algoritmo de resolución del modelo SR fallará en su primer paso, lanzando al agente contra el muro w , independientemente del parámetro D escogido.

La Figura 4.6 muestra la evolución de la región de fallo de una situación concreta a medida que se aumenta la distancia del horizonte D del agente. La imagen apoya de manera gráfica el hecho que prueba el Teorema 4.2. Las subregiones de fallo $F_{D'}$ tienden a un conjunto-límite F que, en este caso particular, parece abarcar una zona abierta del plano (un área no nula); en ella el algoritmo falla irremediabilmente, sea cual sea el alcance visual del agente.

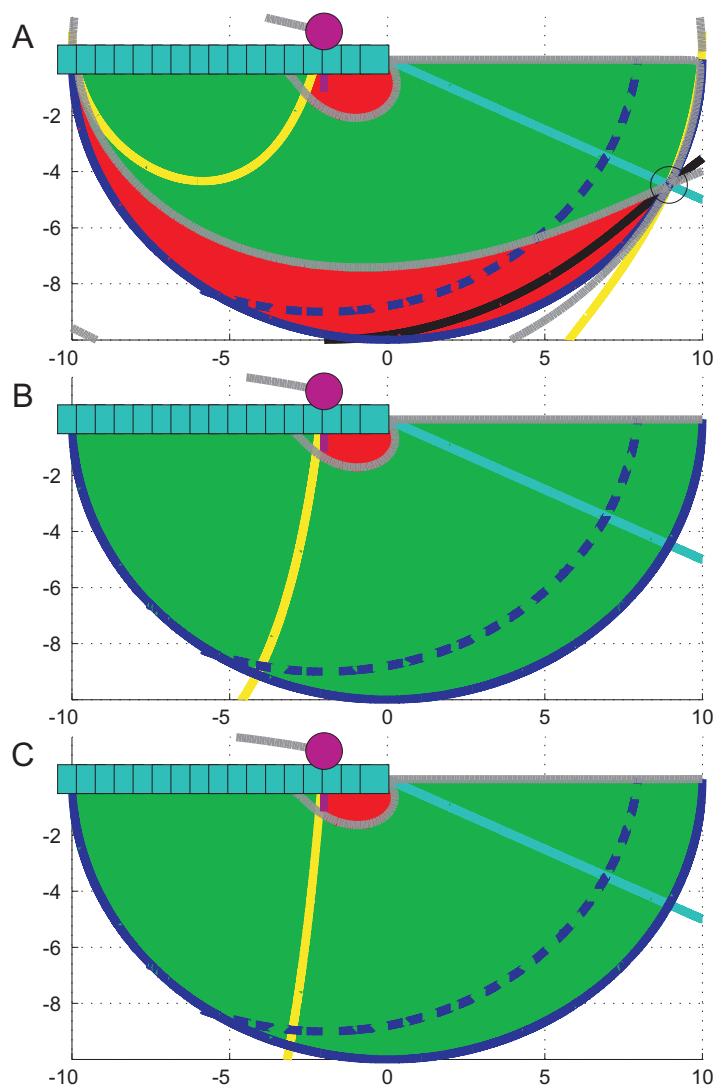


Figura 4.6: Evolución de las regiones de fallo y acierto del algoritmo a medida que se aumenta la distancia del horizonte del agente. En verde, las posiciones iniciales que son resueltas de manera razonable por el primer paso del modelo SR; en rojo, las posiciones para las que el modelo sugiere dirigir al agente contra el muro w . Para las curvas que delimitan estas regiones se ha mantenido el mismo código de color que se usó previamente en las Figuras 4.3, 4.4 y 4.5, añadiéndose en esta ocasión la representación de la frontera de C_D , en trazos azules discontinuos. Los conjuntos $F_{D'}$, para cada D' seleccionado, se corresponden con las zonas rojas situadas por encima de esta línea discontinua. El pequeño segmento color púrpura representa el conjunto de puntos incluidos en F e identificados en la demostración del Teorema 4.2. Se ha partido de la situación recogida en la Figura 4.4: $D = 10$, $(x_T, y_T) = (-2, 1)$. (A) $D' = 10$. (B) $D' = 20$. (C) $D' = 30$.

Capítulo 5

Conclusiones

El estudio del comportamiento humano y, en concreto, de la navegación cognitiva en entornos sociales nos permite entender los mecanismos biofísicos y sociales subyacentes a la toma inteligente de decisiones. Los avances en la modelización matemática de estos mecanismos pueden suponer grandes beneficios en áreas de salud, transporte, así como en el desarrollo de diferentes tecnologías, en general. Una de las aplicaciones más prometedoras hoy en día es la robótica cognitiva; nuevos algoritmos y modelos podrán facilitar la navegación “humana” y, como consecuencia, habilitarán la incorporación de robots a la sociedad.

En este trabajo se ha desarrollado el estudio del algoritmo local de navegación *Simple Rules* (SR, por sus siglas en inglés), propuesto por Moussaid, Helbing y Theraulaz [5]. La filosofía de este modelo responde, esencialmente, a dos heurísticas conductuales básicas, que dotan a los individuos involucrados de autonomía para decidir en cada paso la dirección y velocidad más adecuadas al contexto en que se desenvuelven.

La modelización matemática de este sistema de navegación, descrita con detalle en el segundo capítulo, es relativamente sencilla: para cada agente se obtienen las coordenadas de sus estados sucesivos (posición, velocidad, dirección visual, etc.) resolviendo en cada paso de tiempo un sistema de ecuaciones diferenciales ordinarias acopladas. Una vez expuesta la matemática del modelo se ha procedido a su implementación en Matlab, con el fin de poder analizar de manera empírica los beneficios e inconvenientes de su uso.

Como se ha recogido en el tercer capítulo, SR es capaz de predecir trayectorias individuales y patrones colectivos de movimiento coherentes con estudios experimentales en el campo de la psicología del comportamiento. Su simpleza con respecto a otros modelos de navegación existentes no le resta efectividad a la hora de resolver gran variedad de situaciones dinámicas de relativa complejidad. Se han analizado dos escenarios en los que el modelo ha sido capaz de encontrar rutas factibles para cada uno de los individuos implicados, evitando las colisiones que irremediablemente habrían tenido

lugar en caso de que las situaciones hubiesen sido afrontadas por agentes no cognitivos.

En otros dos escenarios se ha querido comparar el efecto que tiene la implementación de este algoritmo en contextos de aglomeración. Una decisión espontánea emergente que sabe reflejar fielmente el modelo SR es la generación de carriles bidireccionales cuando en una muchedumbre coexisten individuos que desean desplazarse en direcciones opuestas. Otro hecho con el que el algoritmo también es coherente es la influencia que la elevada ocupación de una sala tiene sobre la longitud del viaje que ha de emprenderse para cruzarla. Para probar esto último se han analizado datos numéricos recogidos de simulaciones realizadas para distintos niveles de aglomeración, obteniéndose resultados concordantes con la naturaleza del experimento.

Las ventajas de este algoritmo son evidentes. Sin embargo, la localidad temporal en que se fundamenta, basada en permanentes tomas de decisión, propician la aparición de diversos problemas que desembocan en una navegación deficiente o subóptima. Estos problemas intrínsecos son comunes a todos los métodos locales que persiguen la búsqueda de un óptimo global; en este caso concreto pueden llegar a corromper el buen funcionamiento del algoritmo, devolviendo como óptimas trayectorias que sólo inducen un comportamiento absurdo del sistema, incluso llegando a tratar como imposibles situaciones que tienen fácil solución. Con la finalidad de dar una explicación a este tipo de fallos, en el cuarto capítulo se ha propuesto una familia de escenarios para los que se ha demostrado analíticamente que siempre es posible inducir una solución del algoritmo para nada deseable: algunos de estos fallos pueden evitarse ajustando ciertos parámetros propios del modelo, pero otros tantos se mantienen de manera irremediable.

En concreto, se han enunciado y demostrado dos teoremas. El primero de ellos prueba que, fijado el límite visual del agente implicado (único parámetro libre que interviene en la toma de decisión del modelo), es posible acotar un área no vacía que encierre exclusivamente localizaciones desde las cuales el algoritmo SR devuelve una decisión de navegación equivocada. El segundo resultado es más fuerte; demuestra la existencia de una región no vacía conformada en su totalidad por localizaciones que inducen un comportamiento absurdo del modelo, para cualquier elección del parámetro señalado.

En definitiva, habiendo motivado este trabajo la búsqueda de un modelo alternativo a otro de carácter global mucho más complejo, de naturaleza cognitiva y desarrollado en un estudio previo, una buena línea de investigación sería considerar la posibilidad de incluir los beneficios a corto plazo proporcionados por uno como complemento a la potencia y versatilidad del otro; de este modo nacería un nuevo sistema de navegación inteligente, capaz de afrontar mediante un enfoque mucho más completo las barreras impuestas por situaciones dinámicas cotidianas, dificultades que se desearía supiese abordar con éxito cualquier agente cognitivo en un futuro no muy lejano.

Bibliografía

- [1] Makarov V.A., Villacorta-Atienza J.A. (2011). Compact internal representation as a functional basis for protocognitive exploration of dynamic environments. In *Recurrent Neural Networks for Temporal Data Processing*, Eds.: H. Cardot, INTECH, pp. 81-102.
- [2] Villacorta-Atienza J.A., Makarov V.A. (2013). Neural network architecture for cognitive navigation in dynamic environments. *IEEE Transactions on Neural Network Learning Systems* 24(12), pp. 2075-2087.
- [3] Villacorta-Atienza J.A., Calvo C., Makarov V.A. (2014). Prediction-for-CompAction: Navigation in social environments using generalized cognitive maps. *Biological Cybernetics* (aceptado).
- [4] Villacorta-Atienza J.A., Velarde M.G., Makarov V.A. (2010). Compact internal representation of dynamic situations: Neural network implementing the causality principle. *Biological Cybernetics* 103, pp. 285-297.
- [5] Moussaid M., Helbing D., Theraulaz G. (2011). How simple rules determine pedestrian behavior and crowd disasters. *PNAS* 108, pp. 6884-6888.
- [6] Moussaid M., Helbing D., Garnier S., Johansson A., Combe M, Theraulaz G. (2009). Experimental study of the behavioral mechanisms underlying self-organization in human crowds. *Proceedings of the Royal Society B* 276, pp. 2755-2762.
- [7] Schadschneider A, Klingsch W., Klüpfel H., Kretz T., Rogsch C., Seyfried A. (2009). Evacuation dynamics: Empirical results, modeling and applications. *Encyclopedia of Complexity and Systems Science*, Ed. Meyers R (Springer, Berlin), pp. 3142-3176.