

WUOLAH



vrnk98

www.wuolah.com/student/vrnk98



12561

Relacion 1 IA.pdf

Relacion 1 IA



2º Inteligencia Artificial



Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
UGR - Universidad de Granada

1. Una hormiga artificial vive en un mundo bidimensional cuadriculado y desarrolla un comportamiento que le permite seguir un rastro de feromonas a lo largo de un conjunto de casillas previamente marcadas (el tamaño del rastro es de una casilla). La hormiga ocupa una sola casilla y puede encarar las casillas que se encuentran arriba, a la derecha, a la izquierda y debajo de la posición en la que se encuentra. La hormiga puede llevar a cabo tres acciones: (1) moverse a una celda hacia adelante, (2) girar a la izquierda permaneciendo en la misma casilla, (3) girar a la derecha permaneciendo en la misma casilla y (4) no hacer nada. La hormiga puede percibir si la casilla que tiene delante (en el sentido del movimiento) tiene o no feromona. Utilizando el código del simulador correspondiente (ejercicio1.zip) y los mapas feromona1.map y feromona2.map:
- (a) Especificar un sistema de reglas para controlar el comportamiento de la hormiga en el seguimiento del rastro de la feromona. Suponer que inicialmente la hormiga se encuentra en una casilla en la que se puede percibir el rastro de feromona.
- (b) Resuelva el problema anterior con la restricción de que la hormiga no puede girar en ningún caso más de 180 o desde la posición en la que aparece en cada casilla, pero sí desde cualquier otra. ¿Corrige el problema anterior?
- (c) Modifique el conocimiento necesario para que la hormiga pase por todas las casillas con feromona del mapa feromona2.map.
- (d) Proponga un mapa en donde el comportamiento definido para el apartado anterior no consiga pasar por todas las casillas con feromona.

Una matriz donde la hormiga se situa en una de las casillas, y al lado de la hormiga aparece un rastro de feromonas q debe seguir (TAM ancho no puede ser más de una casilla). hormiga ancho 1, puede ir arriba, abajo, izq, dcha, acciones: avanzar hacia delante, girar dcha/izqda sin mover de casilla o no hacer nada.

- a) Objetivo diseño: seguir rastro de feromonas
 Percepciones: F->{V si hay feromona en casilla encarada, F en caso contrario}
 Acciones: AV->avanzar
 GI->girar izqda en la misma casilla
 GD->girar dcha en la misma casilla
 IDLE->No hacer nada
 Memoria: no hay
 Selección de acciones:
 Sistema de producción:
 F->AV
 ¬F->GD
- b) Memoria: Giro{-1->Ha hecho giro a dcha, 0->No ha hecho giro, 1->Ha hecho giro a izqda, 2->Ha hecho 2 giros a izqda}
 Sistema de producción:
 F->{AV, Giro=0}
 Giro= 0->{GD, Giro=-1}
 Giro=-1->{GI, Giro= 1}
 Giro= 1->{GI, Giro= 2}
 IDLE->No hacer nada
- c) Sistema de producción
 F->{AV, Giro=0}
 Giro= 0->{GD, Giro=-1}
 1->GI
- d) map3

2. La avispa hembra del género *Sphex* deja sus huevos dentro de un grillo que ha paralizado y llevado a su nido. Las larvas de la avispa salen del grillo y se alimentan de él. La avispa presenta el siguiente comportamiento:

La rutina de la avispa consiste en llevar el grillo paralizado a su nido, dejarlo en el umbral (del nido), entrar para ver si todo está correcto, salir, y entonces arrastrar el grillo hacia su interior. Si el grillo se mueve cuando la avispa está en el interior haciendo la inspección preliminar, la avispa saldrá del nido, volverá a colocar el grillo en el umbral, pero no dentro, y repetirá el procedimiento de entrar en el nido para ver si todo está correcto. Si el grillo se mueve otra vez mientras la avispa está dentro del nido, ésta volverá a salir y colocar el grillo en el umbral, entrando de nuevo en el nido para realizar la inspección preliminar. En una ocasión, este procedimiento se repitió cuarenta veces. Diseñar características, acciones y un agente reactivo que se corresponda con el comportamiento de la avispa.

Objetivo diseño: Meter al grillo dentro del nido

Percepciones: $G\{T \text{ hay grillo}, F \text{ no hay grillo}\}$

$L\{D \text{ dentro}, F=\text{Fuera}\}$

$V\{T \text{ ha vigilado dentro}, F \text{ no ha vigilado}\}$

Acciones: CG->Coger al grillo y llevarlo al umbral

IRD->Ir dentro

IRF->Ir fuera

MG->Meter al grillo

Memoria: no hay memoria

Sistema de producción:

$V \wedge G \wedge L = F \rightarrow MG$

$\neg G \wedge L = F \rightarrow CG, V=F$

$\neg V \wedge G \wedge L = F \rightarrow IRD, V=T$

$L = D \rightarrow IRF$

Inicialización: El grillo está lejos y la avispa está fuera sin grillo (poor avispa) y no ha vigilado

3. Supongamos que un agente trabaja sobre un tablero formado por $n \times n$ casillas. Sobre este tablero se definen dos zonas: una “zona interior” y una “zona exterior” formada por el resto de las casillas. Separando ambas zona aparece una línea gruesa denominada “Frontera”. En la figura se muestra un ejemplo de la configuración de un tablero 7×7 .

El cometido del robot consiste en llevar el obstáculo que se encuentren en la zona interior a la zona exterior. El robot siempre se debe encontrar en la zona interior, y no debe nunca traspasar la frontera.

Para realizar esta tarea, el robot dispone de 2 sensores, un sensor de choque “BUMPER” que le permite detectar el obstáculo, un sensor de infrarrojos “CNY70” que permite ver dónde está la línea de la Frontera. Los dos sensores se encuentran situados en la parte frontal del robot.

Las acciones que puede realizar el robot son las siguientes:

- FORWARD: Avanza una casilla en la dirección que marca su brújula siempre que no tenga un obstáculo delante.
- BACKWARD: Retrocede una casilla en la dirección contraria a la que indica su brújula, siempre que no tenga un obstáculo detrás.
- TURN L: Gira sin moverse de la casilla hacia la izquierda.
- TURN R: Gira sin moverse de la casilla hacia la derecha.
- IDLE: No realiza ninguna acción.
- PUSH: Avanza una casilla en la dirección que marca su brújula. Para que esta acción tenga efecto, debe estar activado el sensor de choque.

Se pide que utilizando el simulador proporcionado (ejercicio3.zip):

(a) Definir las variables de estado (nombre e interpretación) y las reglas de producción (expresadas en forma de reglas de producción) necesarias para diseñar un agente reactivo con memoria que partiendo de una casilla desconocida dentro de la zona interior de un tablero cuadrado de dimensiones también desconocidas (nunca superiores a 99×99), sea capaz de calcular la dimensión de la zona interior, suponiendo que en el tablero no hay obstáculos. Probar con los mapas `tablero1.map` y `tablero2.map`.

(b) Definir las variables de estado y las reglas de producción necesarias que permitan al robot localizar el obstáculo en el tablero. Probar con los mapas `tablero3.map` y `tablero4.map`.

(c) Suponiendo que el robot se encuentra orientado hacia el obstáculo en una casilla adyacente (es decir, BUMPER es verdad) y que el obstáculo se encuentra en una casilla interna del tablero que no es adyacente con ninguna casilla pegada a la frontera, definir las variables de estado y las reglas de producción necesarias que permitan al robot expulsar el obstáculo hacia la zona exterior, arrastrándolo por el camino más corto de casillas. Probar con los mapas `tablero5.map`, `tablero6.map`, `tablero7.map` y `tablero8.map`.

Si no hemos pasado, negativo si es frontera, intentamos ir a la casilla que menos veces hayamos pasado d las contiguas que tenemos

Objetivo: sacar al obstaculo

Percepcion: CNY70 ->Borde{T,F}

BUMPER->choque obstaculo{T,F}

Acciones: F->Adelante

B->Atras

GI->giro izda

GD->giro derecha

IDLE-> NAda

PUSH->empujo

Memoria: Brujula{NSEO, punto cardinal al que estoy orientado}

(x,y) posicion en el mapa

Mapa[7][7]= mapa del entorno

Inicializacion:

Brujula N

Posicion (0,0)

Mapa[i][j]=0

Sistema de produccion:

BUMPER \wedge \neg CNY70 -> {Push, actualizar (x,y)}

Si estoy orientado mirando a la casilla q he estado menos numero de veces, avanzo,
si no miro a donde estoy mirando, hago un giro hasta que esté orientado a esa casilla,
si el CNY70 esta activado esa casilla la pongo a -1

4. Supongamos que tenemos un robot sobre un mapa bidimensional discreto de tamaño $n \times m$. El robot puede realizar las acciones de Avanzar, Girar en el sentido de las agujas del reloj o Quedarse quieto. El robot posee un sistema de posicionamiento sobre el mapa que le devuelve sus coordenadas absolutas (robot x , robot y) dentro del mapa.

Suponiendo que en el mapa hay obstáculos fijos (paredes), y que el robot se encuentra ubicado dentro de ese mapa en una posición concreta se pide que utilizando el simulador proporcionado (ejercicio4.zip):

Definir un comportamiento reactivo para el mismo que le permita desplazarse hasta una coordenada objetivo (obj x , obj y). Para ello, definir las variables de estado necesarias y el sistema de reglas de producción que reproducen el comportamiento requerido.

Objetivo: Desplazarse hasta una coordenada objetivo (obj x , obj y)

Percepción: O{V si hay obstáculo, F en caso contrario}

Acciones:

AV → Avanzar

GD → Girar a la derecha

IDLE → No hacer nada

Memoria: No hay

Sistema de producción:

O → TURN

\neg O → Av