

Laboratorio de Robots Móviles

Practica No. 2

Comportamientos Reactivos

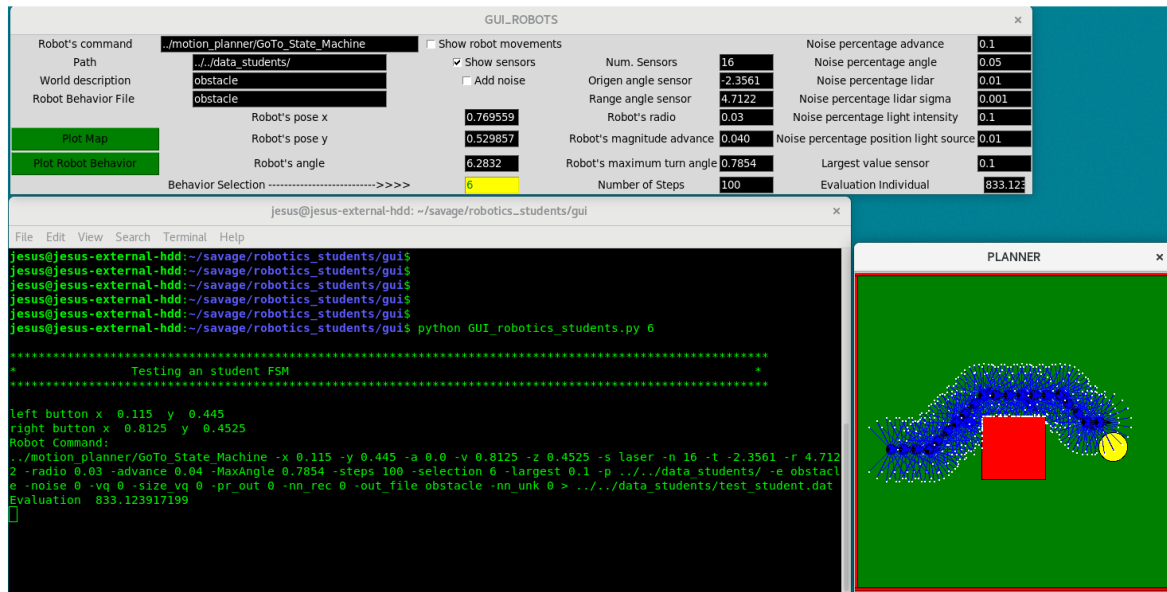
Duración 2 semanas

Objetivo:

- Familiarizar al alumno con los comportamientos reactivos.

Duración: Dos semanas

1. La siguiente figura muestra la interface gráfica que se utilizará para mostrar los comportamientos reactivos de un robot móvil.



Una vez instalado el simulador robotics_students, en el directorio user/robotics_students/gui se encuentra el código en Python de esta interface gráfica.

Para ejecutar esta interface gráfica, teclear: `python GUI_robotics_students.py`, para ver cómo se usa el sistema. Se puede seleccionar un comportamiento específico colocando el número del comportamiento deseado, por ejemplo para ejecutar el comportamiento número dos de los estudiantes teclear:

```
python GUI_robotics_students.py 6
```

En la interfaz gráfica también se puede seleccionar el comportamiento del robot en la opción **Behavior Selection**. Con un 1 se selecciona un comportamiento reactivo sin memoria que se dirige a una fuente luminosa evadiendo obstáculos; con un 2 se selecciona un comportamiento de una máquina de estados que evade obstáculos; con un 3 se selecciona el comportamiento de una máquina de estados que se dirige a una fuente luminosa; con un 4 se selecciona un algoritmo que se dirige a una fuente luminosa evadiendo obstáculos; con un 5 y un 6 se seleccionan los

comportamientos escritos por los estudiantes, originalmente se tienen repetidos el comportamiento 1 y 4 para que el estudiante los pueda modificar.

En la interface gráfica, en la ventana PLANNER, con el botón izquierdo del mouse seleccione la posición inicial del robot y con el derecho la final, después observe el comportamiento del robot. Seleccionando de nuevo el botón derecho del mouse el robot tomará su última posición como origen y el destino la posición del botón derecho. Las coordenadas (0,0) del mapa se encuentran en el lado inferior izquierdo de la figura. Familiarícese con el funcionamiento de la interfaz gráfica con diferentes configuraciones y comportamientos.

En la interface gráfica en el campo “**World description**” se puede seleccionar el entorno donde opera el robot simulado, hay 14 entornos: obstacle, random_1, random_2,..., random_13. Cuando se selecciona un nuevo entorno, presione el botón “**Plot Map**” para mostrarlo. En el apéndice A se muestra un mapa simbólico en donde el robot navega, los objetos se representan con polígonos. Este tipo de archivos tienen terminación *.wrl y se encuentran en el directorio: user/data_students.

El resultado de la navegación del robot se puede ver nuevamente presionando el botón “**Plot Robot Behavior**”. Para mostrar los sensores del robot seleccione el check box de “**Show sensors**”. Para mostrar el movimiento del robot paso a paso seleccione el check box de “**Show robot movements**”. Para agregar ruido a los sensores y al movimiento, seleccione check box de “**Add Noise**”.

2. En el directorio user/robotics_students/motion_planner se encuentra el código GoTo_State_Machine.cpp, el cual ejecuta los algoritmos de navegación del robot de acuerdo del comportamiento seleccionado. Este programa es llamado por la interface gráfica pasando los parametros de ejecución de acuerdo al comportamiento seleccionado. GoTo_State_Machine.cpp usa las máquinas de estado que se encuentran en el directorio: /robotics_students/state_machines Familiarícese con el código que se encuentra en GoTo_State_Machine.cpp y con el código de las máquinas de estados.

3. Modifique el código de user/robotics_students/state_machines/state_machine_student.h para que el robot tenga un comportamiento que se dirija a una de las paredes del medio ambiente random_1 y después la empiece a rodear. Pruebe su comportamiento usando un comportamiento reactivo sin memoria y uno con memoria usando el concepto de máquina de estados. Para compilar su código, habiendo editado los códigos GoTo_State_Machine.cpp y state_machine_student.h, hacer lo siguiente: *colocarse en el directorio* user/robotics_students/motion_planner y dar el siguiente comando: make

4. Programe el comportamiento Bug 1 en el medio ambiente “obstacle”, después con cualquiera de los ambientes denominados randoms. Pruebe su algoritmo ya sea sin ruido o con ruido en el movimiento y sensado del robot, para agregar ruido a la simulación seleccione el check box **Add Noise** .

APENDICE A

Mapa Simbólico obstacle.wrl

En este apéndice se muestra las siguientes directivas de la representación de los mapas simbólicos:

1. Se pueden incluir comentarios o comentar una línea si se coloca un “;” al principio del renglón.

```
; * File: room.wrl *
```

2. “dimensions” indica las dimensiones del medio ambiente, las cuales están indicadas en el sistema métrico decimal. En el siguiente ejemplo se presentan las dimensiones del medio ambiente “room” de 1m x 1m:

```
( dimensions room 1.000 1.000 )
```

3. “polygon” indica los vértices de un polígono, se indica si es de tipo obstáculo o pared, después viene el nombre del obstáculo y a continuación los vértices. Los vértices de los polígonos se indican con las coordenadas Xi y Yi, ordenados hacia el sentido horario de las manecillas del reloj:

```
( polygon obstacle obs1 .40 .55 .60 .55 .60 .35 .40 .35 )
```

```
( polygon wall wall1 0.0 0.0 0.0 1.0 0.01 1.0 0.01 0.0 )
```

obstacle.wrl

```
; *****  
; * File: room.wrl *  
; * Definition of the forbidden and allowed areas in the Robot's *  
; * world. These areas are derivated from the objects in the *  
; * world. *  
; *****  
( dimensions room 1.000 1.000 )  
( polygon obstacle obs1 .40 .55 .60 .55 .60 .35 .40 .35 )  
( polygon wall wall1 0.0 0.0 0.0 1.0 0.01 1.0 0.01 0.0 )  
( polygon wall wall2 0.0 0.99 0.0 1.0 1.0 1.0 1.0 0.99 )  
( polygon wall wall2 0.99 1.0 1.0 1.0 1.0 0.0 0.99 0.0 )  
( polygon wall wall2 1.0 0.01 1.0 0.0 0.0 0.0 0.0 0.01 )
```