

ROBÓTICA
CUARTO CURSO DEL GRADO EN
INGENIERÍA INFORMÁTICA



Práctica 7

ROS: Maestro Remoto y TURTLEBOT Real

F. Gómez Bravo
R. López de Ahumada
José Manuel Lozano

En esta práctica se ilustra como realizar la configuración de ROS para acceder a los topics gestionados por nodos que se ejecutan en otra máquina. Igualmente se detalla la forma en que se utilizará este procedimiento para programar y controlar el robot turtlebot real.

1. Introducción

Una de las ventajas de ROS es que permite trabajar siguiendo la filosofía de un sistema distribuido, haciendo posible repartir la carga de trabajo entre diferentes equipos. Para ello, es necesario definir un nodo 'roscore' como maestro, y configurar en el resto de máquinas la dirección IP (o el nombre asociado a esa dirección IP) donde se encuentra ejecutando dicho nodo.

Con este fin, modificaremos el fichero '/etc/hosts' de forma que incluyamos el nombre de todas las máquinas que controlan los robots y su correspondiente dirección IP.

El texto a incluir debe ser el siguiente:

192.168.2.58 MrBarret

192.168.2.59 Ernest

192.168.2.60 Castor

192.168.2.61 Pollux

192.168.2.62 Getrud

Para configurar el 'roscore' maestro en otro ordenado operaremos según se describe a continuación.

Supongamos, por ejemplo, que desde el ordenador de sobremesa se desea acceder a los servicios del 'roscore' de la máquina MrBarret. Para ello abriremos un terminal en el ordenador de sobremesa y ejecutamos:

```
turtlebot@rosindigo:~$ export ROS_MASTER_URI=http://MrBarret:11311  
  
turtlebot@rosindigo:~$ export ROS_HOSTNAME=Dirección IP del computador local
```

Ahora, cualquier nodo que ejecutemos en la ventana del terminal tendrá acceso a la información de los topics que gestione el 'roscore' de la máquina Mr Barret.

Tendremos que ejecutar estas líneas de comando cada vez que abramos un terminal para ejecutar un nodo que interaccione con el 'roscore' remoto.

2. Interacción con el robot Real

Para explicar más claramente el procedimiento, vamos a describir un ejemplo en el que suponemos que se interactuará con la máquina MrBarret.

En primer lugar, vamos a descargar el nodo programado en el fichero 'control_joy.py' en el directorio del robot: '/home/turtlebot/catkin_ws/src/control_turtlebot/scripts'. Para ello vamos a hacer uso del programa Filezilla. Buscamos este programa en lanzador de aplicaciones, lo seleccionamos y lo ejecutamos. Una esté en ejecución, lo configuramos para establecer una conexión 'sftp' con la máquina remota, según se muestra en la figura 1.

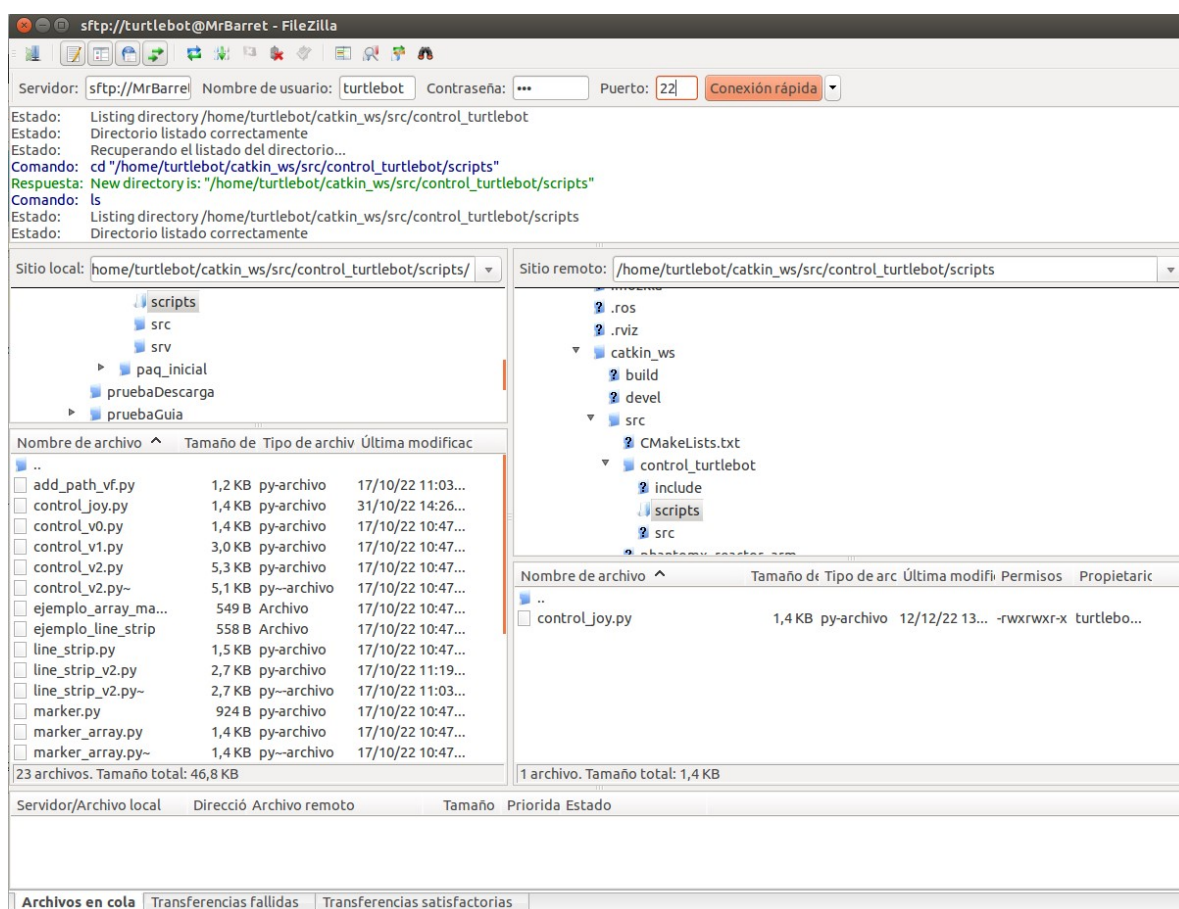


Figura 1

Posteriormente, utilizamos Filezilla para para descargar el fichero en el directorio que hemos especificado antes.

Una vez hecho esto, será necesario ejecutar este nodo en la máquina MrBarret. Para ello, abriremos un terminal y estableceremos con ella una conexión ssh de la siguiente forma:

```
turtlebot@rosindigo:~$ ssh turtlebot@MrBarret
```

Cuando nos encontremos conectados a la máquina remota, arrancaremos un 'roscore'.

A la vez, habremos abierto otra ventana terminal, donde tras ejecutar de nuevo un 'ssh' para entrar en la máquina remota, nos situaremos en el directorio donde está contenido el fichero 'control_joy.py' y lo transformaremos en un fichero ejecutable.

A partir de aquí, ejecutamos el nodo 'control_joy' que hemos ubicado dentro del paquete 'control_turtlebot'

```
roslaunch control_turtlebot control_joy.py
```

¡¡¡Pero OJO!!! En esta ocasión el nodo se está ejecutando en el ordenador remoto (es decir en MrBarret).

A continuación, vamos a configurar todo de manera que el nodo que hemos arrancado en MrBarret reciba datos de un nodo que se ejecuta en el computador de sobremesa. Para ello, trabajaremos en la misma ventana en la que hemos definido MrBarret como ROS_MASTER. A partir de aquí, seguiremos los pasos descritos en la práctica 6 para la utilización del joystick en ROS. Es decir, averiguaremos el puerto al que se conecta el joystick y lo configuramos como puerto de lectura para el nodo 'joy_node'. Una vez hecho esto, ejecutamos el nodo 'joy_node' utilizando el comando roslaunch.

¡¡¡Pero OJO!!! En esta ocasión el nodo se está ejecutando en el ordenador de sobremesa.

Después, abriremos otra ventana y otra comunicación 'ssh' con el computador remoto. Ahora utilizaremos el comando 'rostopic' con la opción 'echo' para visualizar la salida del 'topic' '/mobile_base/commands/velocity'. Debemos comprobar que cada vez que movemos el joystick conectado al ordenador de sobremesa, se publican datos en el 'topic' correspondientes al control de la velocidad del robot.

Tras verificar que todo funciona correctamente, pasaremos a comprobar cómo ahora es posible controlar el robot desde el ordenador de sobremesa y visualizarlo en el mismo computador.

Con este fin, debe abrir otra ventana de comandos y establecer una nueva comunicación 'ssh'. Una vez hecho esto, ejecutamos el comando lanzador:

Para MrBarret, Ernest y Gertrude:

```
roslaunch turtlebot_bringup minimal.launch
```

Para Castor y Pollux:

```
roslaunch kobuki_node robot_with_tf.launch
```

Con esto hemos puesto en marcha el controlador de bajo nivel del robot que convierte los comandos de velocidad lineal y angular en señales de control para la velocidad del giro de las ruedas. Ahora el robot debe moverse según se actúe el 'jostick'. Pruebe a moverlo de una manera suave y prudente

Finalmente, será posible visualizar el movimiento del robot desde el ordenador de sobremesa abriendo una ventana de comando y ejecutando el nodo 'rviz'.

Antes, realizaremos el mismo procedimiento descrito al principio, para colocar el fichero 'add_path_vf.py' en el directorio '/home/turtlebot/catkin_ws/src/control_turtlebot/scripts', y configurarlo como fichero ejecutable. Una vez hecho esto, es necesario abrir una ventana de terminal y ejecutar de nuevo los comandos.

```
turtlebot@rosindigo:~$ export ROS_MASTER_URI=http://MrBarret:11311  
turtlebot@rosindigo:~$ export ROS_HOSTNAME=Dirección IP del computador local
```

A partir de aquí ejecutamos 'rviz' en el ordenador de sobremesa.

```
turtlebot@rosindigo:~$ rosrun rviz rviz
```

Finalmente, tal y como se hizo en la práctica 6, elegimos 'odom' como sistema de referencia global, y añadimos un objeto TF correspondiente al robot y un objeto 'path' correspondiente a la trayectoria descrita por el robot.

Ahora podrá mover el robot con el 'joystick' y visualizar el movimiento del mismo, todo desde el ordenador de sobremesa.

3. Tarea

Utilizando lo desarrollado en la práctica anterior y lo aprendido en esta, configure el robot real para que el robot siga un camino predefinido. Visualice el robot en todo momento utilizando el ordenador de sobre mesa.