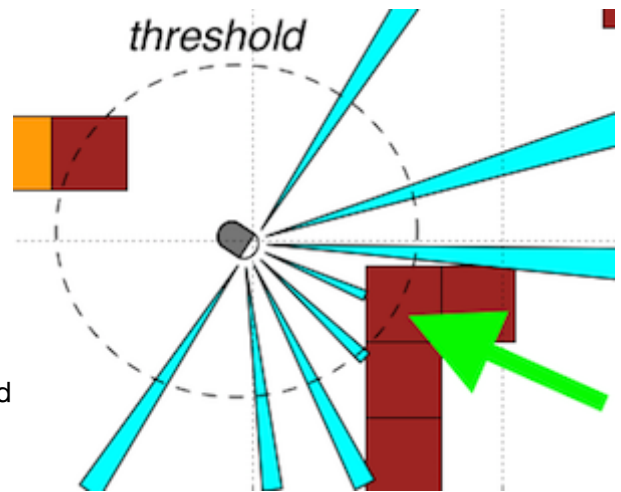


Detecting Obstacles

An obstacle can be detected by comparing the values returned by the ultrasonic sensor with a predefined **distance threshold**.

For values below that threshold, the detected obstacle is considered too close to the robot, and an action should be taken, for example stopping and/or turning, in order to avoid collision.

In the example figure, the value of sensor 3 is less than the threshold (represented by the dotted circle), as signaled by the green arrow.



In [1]:

```
import packages.initialization
import pioneer3dx as p3dx
p3dx.init()
```

Exercise:

Make a program for the robot to move forward until any of the front sensors (numbered 3 and 4) detects an obstacle below a given distance threshold, for example 1 meter. Use the following template code:

In [3]:

```
threshold = 1 # in meters
...
while p3dx.distance[3] > threshold and p3dx.distance[4] > threshold:
    p3dx.move(2.5, 2.5)
p3dx.stop()
```

You may confirm the proper functioning of your code with the following test:

In [4]:

```
for i in range(3,5):
    print("Sensor %d: %.3f m" % (i, p3dx.distance[i]))
```

Sensor 3: 0.950 m

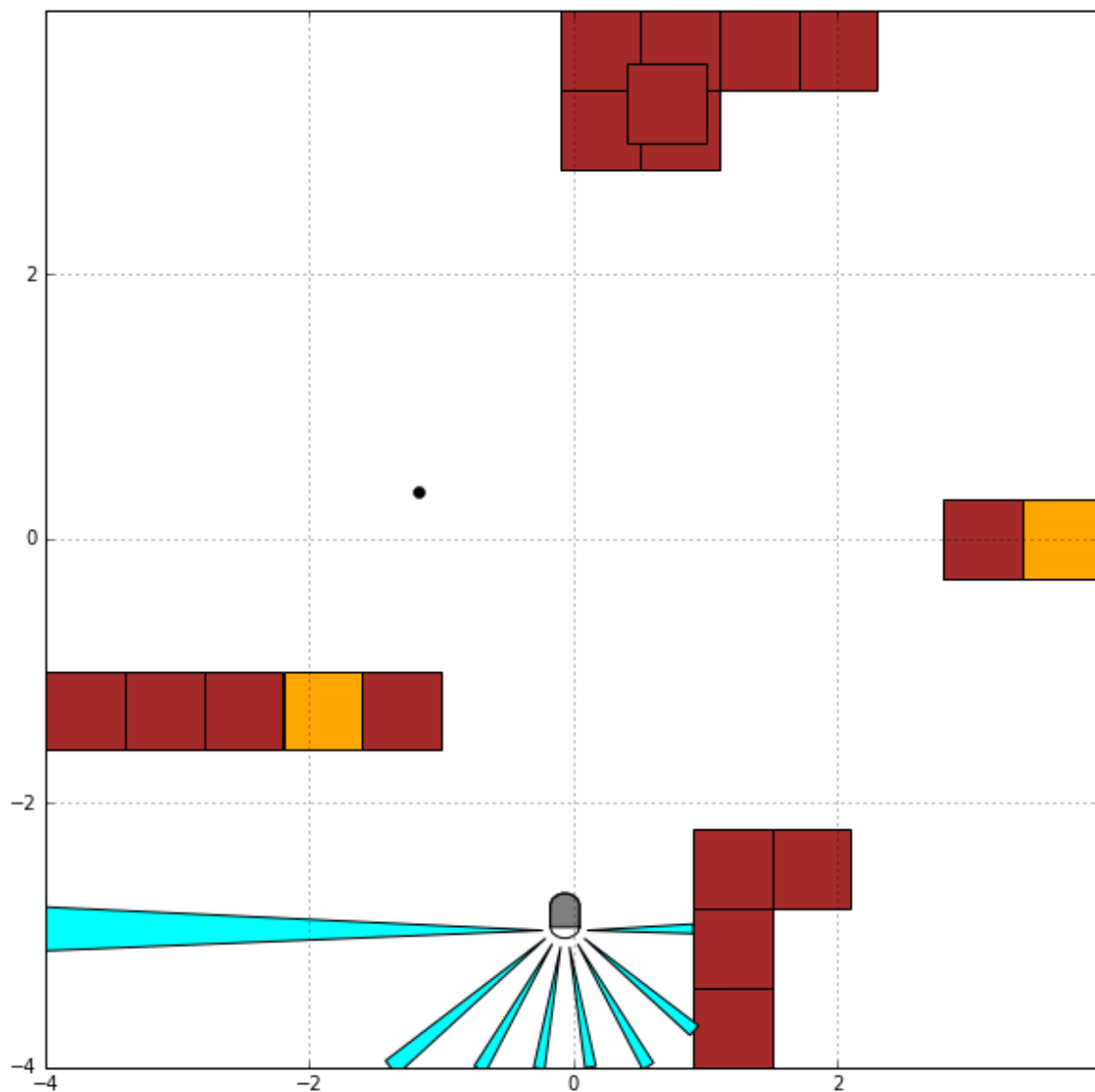
Sensor 4: 0.950 m

And you can plot a diagram with the position of the robot in the environment, and the measurements of the ultrasonic sensors.

The plot is not updated automatically as the robot moves. You need to click on the "Refresh" button.

In [5]:

```
%matplotlib inline
import ultrasonic
ultrasonic.plot()
```



Let's try the next step: [searching for free space \(Searching%20Space.ipynb\)](#).

Try-a-Bot: an open source guide for robot programming

Developed by:



<http://robinlab.uji.es>

Sponsored by:



<http://www.ieee-ras.org>



<http://www.cyberbotics.com>



<http://www.theconstructsim.com>

Follow us:



<https://www.facebook.com/RobotProgrammingNetwork>



<https://www.youtube.com/user/robotprogramming>
