4. Obstacle avoidance via bubble rebound algorithm

4.1 Obstacle detection

The following reactive algorithm for real time obstacle avoidance is described in [1].
We consider a robot with a ring of N equidistant ultrasonic sensors in the front, covering an angle of 180° as shown below.

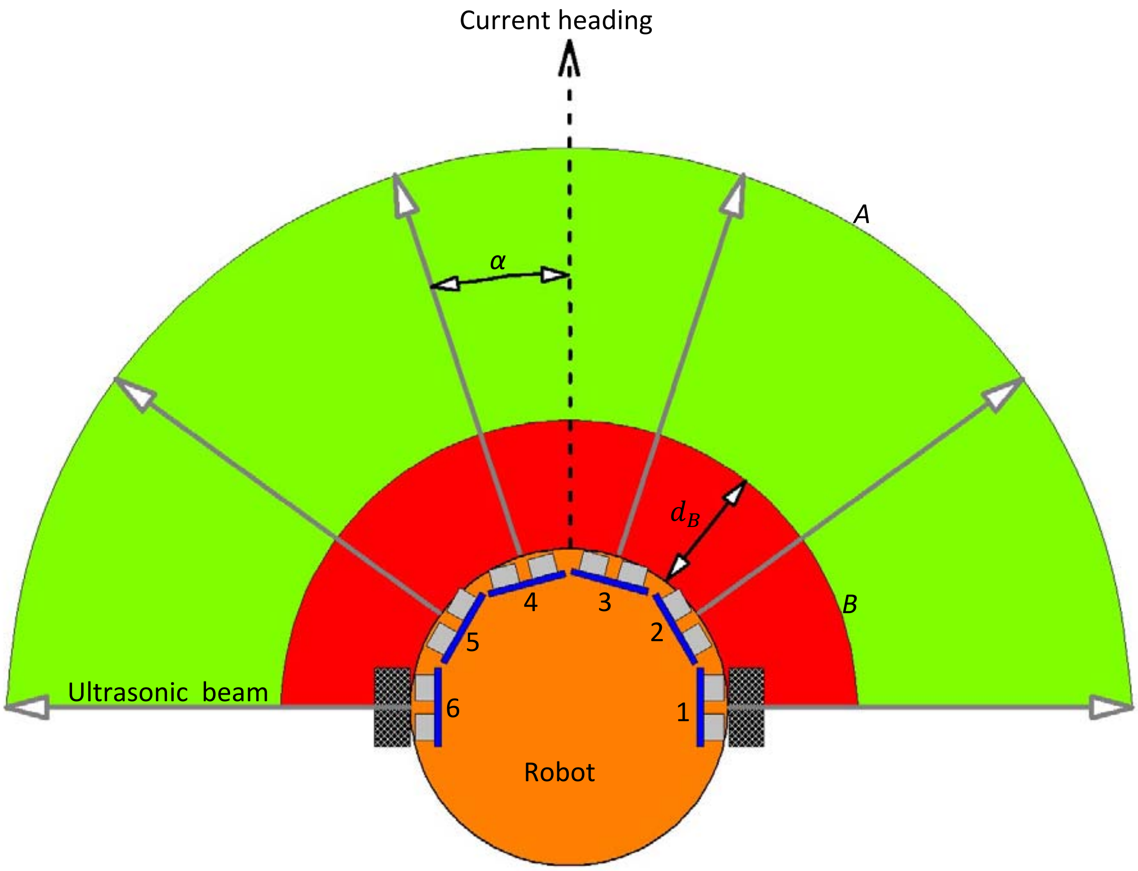

Fig. 4.1.1 Ultrasonic sensor array with sensitivity bubble

Let *A* be the ultrasonic range boundary (in our case 400 cm) and *B* the sensitive bubble boundary. In the simplest case (if the robot drives with invariant velocity and the time intervals between successive evaluations of sensor data are equal), the distance of the ultrasonic sensors to the bubble boundary is a constant, called d_B. If the measured distance d_i of an ultrasonic sensor i to an object is ≤ d_B, the obstacle avoidance algorithm starts, else not and the robot continue to drive in current direction.

4.2 Obstacle avoidance algorithm

Initially, the robot moves straight towards the goal. If an obstacle is detected within the sensitivity bubble, the robots "rebounds" in a direction found as having the lowest density of obstacles, and continues its motion in this new direction until the goal becomes visible (i.e. no obstacle within the visibility range of the sonar in that direction), or until a new obstacle is encountered.
Since the ultrasonic sensors are uniformly distributed, covering an arc of 180°, the sonar readings can be represented in a polar diagram, as shown in figure 4.2.1.

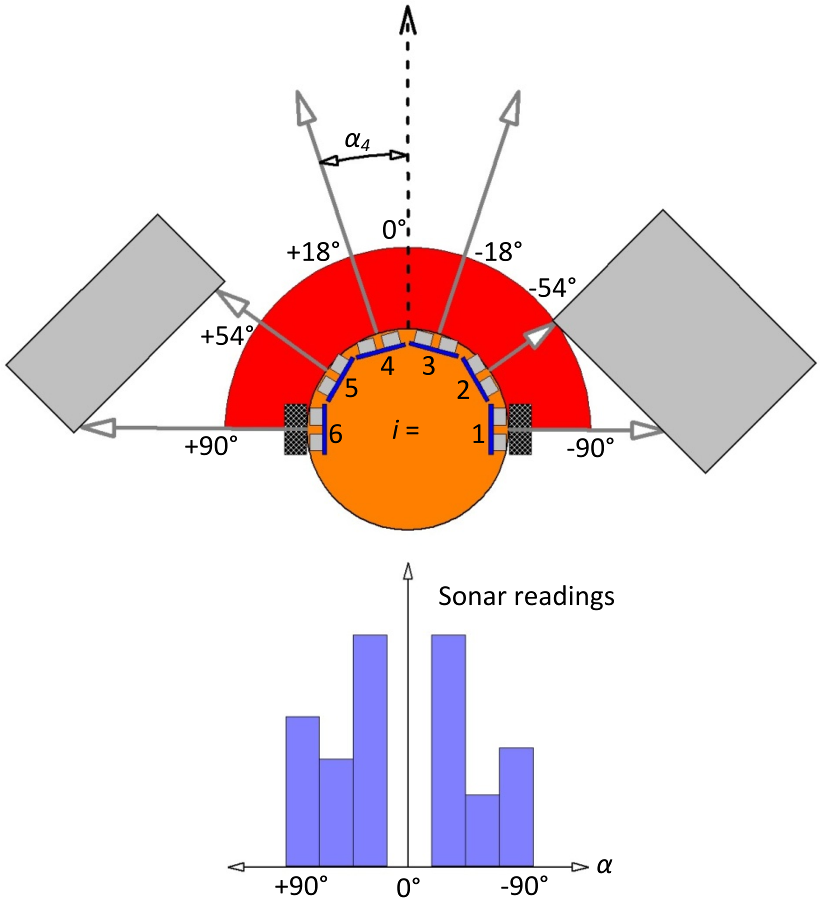

Fig. 4.2.1 Polar diagram of sonar readings

4.3 Computing rebound angle

The rebound angle is computed by weighted arithmetic mean. The weighted mean is similar to the arithmetic mean where instead of each of the data points contributing equally to the final average, some data points contribute more than others. Weighted arithmetic mean is computed by using following formula:

(4.3.1) $$\bar{x} = \frac{\sum_{i=1}^{N} w_i \cdot x_i}{\sum_{i=1}^{N} w_i}$$

where:

$\bar{x}$ is the weighted arithmetic mean
w is the weight
x is the data

In our case the weighted arithmetic mean is the rebound angle α_R, the data are the sonar sensor angles α and the weights are the distance values d, reported by the sonar sensors. Thus

(4.3.2)

$$\alpha_R = \frac{\sum_{i=1}^{N} \alpha_i \cdot d_i}{\sum_{i=1}^{N} d_i}$$

It can be easily noticed that $\alpha_1 = -\alpha_6$, $\alpha_2 = -\alpha_5$ and so on. Using six sonar sensors as in fig. 4.2.1 and substituting the sonar sensor angles accordingly, the equation (4.3.2) becomes

(4.3.3)

$$\alpha_R = \frac{\alpha_6 \cdot d_6 + \alpha_1 \cdot d_1 + \alpha_5 \cdot d_5 + \alpha_2 \cdot d_2 + \alpha_4 \cdot d_4 + \alpha_3 \cdot d_3}{d_1 + d_2 + d_3 + d_4 + d_5 + d_6}$$

$$\alpha_R = \frac{\alpha_6 \cdot d_6 - \alpha_6 \cdot d_1 + \alpha_5 \cdot d_5 - \alpha_5 \cdot d_2 + \alpha_4 \cdot d_4 - \alpha_4 \cdot d_3}{d_1 + d_2 + d_3 + d_4 + d_5 + d_6}$$

$$\alpha_R = \frac{\alpha_6 \cdot (d_6 - d_1) + \alpha_5 \cdot (d_5 - d_2) + \alpha_4 \cdot (d_4 - d_3)}{d_1 + d_2 + d_3 + d_4 + d_5 + d_6}$$

To turn the robot precisely into rebound angle position, a compass module will be used.

References

[1] IOAN SUSNEA, VIOREL MINZU, GRIGORE VASILIU, " Simple, Real-Time Obstacle Avoidance Algorithm for Mobile Robots", ISBN: 978-960-474-144-1