IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 2017

BEng Honours Degree in Computing Part III
BEng Honours Degree in Electronic and Information Engineering Part III
MEng Honours Degree in Electronic and Information Engineering Part III
BEng Honours Degree in Mathematics and Computer Science Part III
MEng Honours Degree in Mathematics and Computer Science Part III
MEng Honours Degrees in Computing Part III
MSc in Computing Science (Specialist)
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the Associateship of the City and Guilds of London Institute

PAPER C333

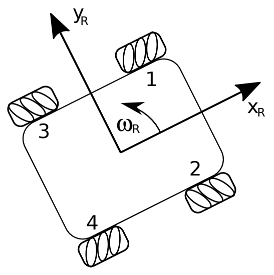
ROBOTICS

Wednesday 14 December 2016, 10:00
Duration: 120 minutes

Answer THREE questions

Paper contains 4 questions
Calculators not required

1 a The omnidirectional robot shown has four drivable wheels, each with diagonal free running rollers embedded in its rim. The robot coordinate frame R is defined such x_R points to the front of the robot, and y_R to its left.



By setting various combinations of the forward angular velocities $\omega_1, \omega_2, \omega_3, \omega_4$ of its wheels, it is able to achieve holonomic motion. Explain what this means, making the contrast with non-holonomic robots with examples. State what combinations of wheel angular velocities it can use to move straight forward in the x_R direction, slide left in the y_R direction, or rotate on the spot in the ω_R direction.

b The inverse kinematics equations of the robot, which relate a desired velocity v_{Rx}, v_{Ry} and angular velocity ω_R of the robot in the robot frame to the wheel angular velocities which need to be set, are set out below. Note that r, l_x and l_y are the known wheel radius, half-width and half-length of the robot respectively; assume that their values are available as progam constants.

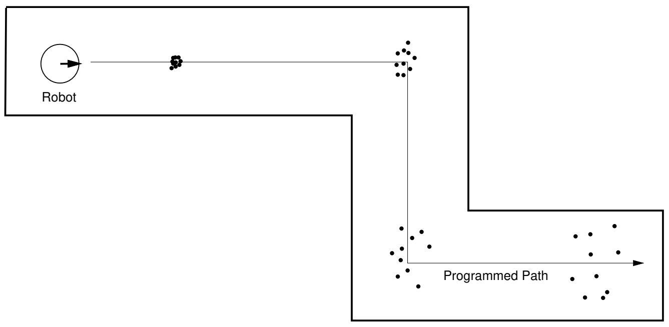
$$\begin{aligned} \omega_1 &= \frac{1}{r}(v_{Rx} - v_{Ry} - (l_x + l_y)\omega_R) & \omega_2 = \frac{1}{r}(v_{Rx} + v_{Ry} + (l_x + l_y)\omega_R) \\ \omega_3 &= \frac{1}{r}(v_{Rx} + v_{Ry} - (l_x + l_y)\omega_R) & \omega_4 = \frac{1}{r}(v_{Rx} - v_{Ry} + (l_x + l_y)\omega_R) \end{aligned}$$

The robot is equipped with a laser sensor which can measure the horizontal angle α in radians to a target object, where α is zero if the object is straight in front, positive if to the left and negative if to the right. In Python pseudocode, write a program which controls the robot to drive forwards steadily at velocity u, without sliding sideways, while controlling its orientation using proportional control to drive towards the target object. The current sensor measurement can be obtained using the function `GetAlpha()` and and the angular velocity of each wheel can be set with `SetOmega(i, omega)`, where `i` is the wheel number.

c The laser sensor is also able to return the current distance D to the target object using the function `GetD()`. Write a a second program which uses proportional control in order that the robot, without rotating, moves towards and holds a position where the target object is a desired distance R directly in front of it.

The three parts carry, respectively, 30%, 40%, and 30% of the marks.

2a A robot is placed in an initially known location on the floor of the corridor shown for which it holds a prior map. It is programmed to drive down the corridor's centre line via straight segments and right angle turns. The robot has a differential drive configuration and relatively uncertain odometry, which means that with no outward-looking sensing a particle distribution representing uncertainty in its position evolves as in the diagram as the robot moves:



The robot now repeats the motion but using a sonar sensor to correct its position estimate via Monte Carlo Localisation. The sonar sensor has a low and constant level of depth measurement uncertainty, and a maximum range which is longer than the corridor. In each of the following cases, copy the diagram and sketch how a distribution of 10 particles would evolve if:

i) The depth sensor is used in a forward-facing configuration.

ii) The depth sensor is used in a sideways-facing configuration.

iii) Two depth sensors are supplied and one faces forwards, one sideways.

b The robot is now equipped with a fixed rig of 12 sonar sensors, mounted horizontally around a circular ring of radius R and facing horizontally outwards at equally spaced angles. The sensors are labelled $i = 0 \dots 11$. Sonar 0 points straight forward, with sonar 1 to its left and the subsequent sonars mounted in label order.

The robot's 2D pose is specified with position and orientation (x, y, θ) relative to a standard coordinate frame. The mapped area's walls are defined with a list of pairs of coordinates (A_x, A_y) and (B_x, B_y) which specify their endpoints. You are provided with the expression:

$$m = \frac{(B_y - A_y)(A_x - x) - (B_x - A_x)(A_y - y)}{(B_y - A_y)\cos\theta - (B_x - A_x)\sin\theta}$$

for the forward distance to an infinite wall passing through (A_x, A_y) and (B_x, B_y) from the centre of a robot with pose (x, y, θ).

Write in Python-like pseudocode a function `GetLikelihood(particle, z, walls)` which returns a floating point unnormalised likelihood value for a particle given a set of sonar measurements `z`. Here `z` is an array of measurements from the 12 sensors, each of which can be accessed as `z[i]`. Each sonar has a Gaussian depth measurement uncertainty, with standard deviation 2cm. Assume that `particle` is an object of the `Particle` class with data members `Particle.x`, `Particle.y` and `Particle.theta`, and that the walls are represented by a Python list `walls` of `Wall` objects, each of which has data members `Wall.Ax`, `Wall.Ay`, `Wall.Bx` and `Wall.By`. Use centimetre and radian units throughout.

The two parts carry, respectively, 30%, and 70% of the marks.

3 a A robot has become lost in a tall office building with 10 floor levels, $i = 1 \ldots 10$. It is equipped with an altitude sensor based on air pressure which has been calibrated to report the floor which the robot is mostly likely to be on: the measurement z from the sensor is a discrete floor number in the range 1 to 10. The sensor, however, does not work perfectly. In earlier tests, it was found that if the robot is truly on a certain floor i, there is a 60% probability that the sensor will output i as its measurement. There is a 13% probability that it will report a floor which is one too high, and another 13% that it will report one too low. Finally, with 14% probability it will report a 'garbage' value which is more than one level away from the true value, equally likely to be any of the other floors.

When initially lost, the robot knows only that it has probability 0.75 of being in the top half of the building (floors 6–10), and 0.25 of being in the bottom half (floors 1–5). It then uses its altitude sensor once and receives the measurement $z = 6$. Using Bayesian probabilistic inference, update a discrete prior probability distribution $P(i)$ over the robot's possible floor level to a posterior distribution $P(i|z)$. Calculate the posterior probability for each floor to two decimal places. Also draw both the prior and posterior distributions as discrete bar charts, with 10 bars in each case to represent the probability per floor.

b A different robot is provided with a new depth sensor, whose performance is initially unknown. Experiments are performed where the sensor is placed at a set of ground truth distances v centimetres from a wall, and in each case 10 readings are taken from the sensor as follows:

Ground truth v (cm)	20.00	40.00	60.00	80.00	100.00
Readings z (cm)	27.72	48.22	65.56	81.63	117.09
	28.28	50.02	66.78	86.32	104.45
	33.65	52.53	77.50	80.05	110.33
	32.18	47.61	80.14	100.61	104.65
	32.69	48.82	76.84	80.31	93.85
	29.18	49.27	65.32	76.01	112.48
	29.87	44.50	73.36	78.09	97.79
	27.65	53.56	60.12	86.01	106.66
	29.07	51.94	59.35	96.81	93.37
	28.73	48.72	76.76	91.40	95.46

Without precise calculation, analyse this data and suggest a straightforward formula for the likelihood function $P(z|v)$ which describes the sensor's performance, including the values of appropriate numerical constants and explaining your reasoning. Bear in mind that the sensor may have systematic as well as zero mean errors; your likelihood function should take this into account.

The two parts carry, respectively, 60%, and 40% of the marks.

4 A robot is equipped with a scanning sonar sensor which can be rotated to point at a horizontal angle α in degrees relative to the robot's forward direction ($\alpha = 0$ when the sonar points forward, and α is positive when it points to the left). The robot uses measurements z in cm from the sonar to update an occupancy map of its surroundings using the log odds form. Assume that the occupancy map has cell size 1cm and covers the area 0cm $\leq x <$ 500cm and 0cm $\leq y <$ 500cm.

a Write Python-like pseudocode for a function `UpdateOccupancyMap(x, y, theta, z, alpha)` to update the occupancy map in response to a single sonar measurement z when the robot is at location (x, y, θ) and the sonar is at angle α. The occupancy map has been pre-allocated globally as a 2D array and the log odds value of the cell at centimetre coordinates (x, y) is stored at `occupancyMap[x][y]`. Assume that the robot is located somewhere near the middle of the occupancy map and that the maximum range of the sonar is less than the distance to the occupancy map's boundaries. The half width of the sonar beam is $5°$ and the standard deviation of its depth measurements is 4cm. Choose sensible values for the amount of increase or decrease in the log odds values at appropriate cells of the occupancy map grid.

b After some time exploring, the robot has created a high quality occupancy map of a room where the occupancy values of all cells within the reachable area have converged to confident values.

Explain with sketches how it would be possible, without needing to make any more measurements from the robot, to generate from the occupancy map a set of location signatures suitable for later place recognition (global localisation) of the robot within the room.

The two parts carry, respectively, 70%, and 30% of the marks.