Solutions to the Written Exam in Mobile Intelligent Systems

Ola Bengtsson
Halmstad University, 2004.05.25

Assistant aids:	Writing tools, Calculator and an arbitrary book on formulas (e.g. Beta)
Location / Date:	Room R1122, Halmstad / 2004-05-25
Time limit:	13:30 – 17:30 => 4 hours
Answers:	All answers should be motivated.
Language:	Write your answers in either **Swedish** or **English** language.
Contact:	Ola Bengtsson, 035 – 167485 or 0733 – 121281
Parts:	**Part I**: Basic questions: Maximum points = 28 [11.0 – 16.5]p gives grade = 3 [17.0 – 22.0]p gives grade = 4 [22.5 – 28.0]p gives grade = 5 **Part II**: Deeper understanding of a scientific paper Maximum points = 18 [07.5 – 10.5]p gives grade = 3 [11.0 – 14.0]p gives grade = 4 [14.5 – 18.0]p gives grade = 5
Passing the exam / Final grade:	You should, to pass the exam, at least have the grade 3 on both individual parts, i.e. you have to pass both parts. The grade is then given as the average of the two individual grades.

Good luck,

/Ola.

Part I: Solution to the Basic Questions

1.1) *A) By simple making more measurements and taking the average value as Y will give a smaller variance of Y than that of X, e.g. taking the average of two measurements yields a variance of:*

$$\sigma_y^2 = \left(\tfrac{1}{2}\right)^2 \sigma_x^2 + \left(\tfrac{1}{2}\right)^2 \sigma_x^2 = \frac{\sigma_x^2}{2}$$

The more measurements used the smaller variance.

B) The standard deviation of Y should be 10 times smaller than the one for X, which means the variance of Y should be $10^2 = 100$ times smaller than the variance of X. To achieve this we have to use N measurements, i.e. the variance of Y can be written as:

$$\sigma_y^2 = \left(\tfrac{1}{N}\right)^2 N\sigma_x^2 = \frac{\sigma_x^2}{N}$$

This should be equal to $\frac{\sigma_x^2}{100}$ which means we have to use 100 measurements.

1.2) *A) The linear combination we are looking for can be written as:*

$$Y = a_1 X_1 + a_2 X_2 \qquad (E.1)$$

The total contribution of X_1 and X_2 should sum up to 1, i.e.

$$a_1 + a_2 = 1 \qquad (E.2)$$

Using Equation 1 and Equation 2 gives the following expression for Y:

$$Y = a_1 X_1 + (1 - a_1) X_2 \qquad (E.3)$$

The variance of Y becomes:

$$Var\{Y\} = Var\{a_1 X_1 + (1-a_1) X_2\} = a_1^2 Var\{X_1\} + (1-a_1)^2 Var\{X_2\} = a_1^2\sigma_1^2 + (1-a_1)^2\sigma_2^2 \quad (E.4)$$

Differentiating with respect to a_1 and setting the result equal to 0 gives the constant a_1, which then gives a_2 as:

$$a_1 = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2} \quad \textit{and} \quad a_1 = \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2}$$

B) With $\sigma_1^2 = \sigma_2^2$ we have Y as the average value of X_1 and X_2, which is logical as both measurements are equally good. $\sigma_1^2 << \sigma_2^2$ means that the measurement from system 1, i.e. X_1, is more reliable than that of X_2, which means that X_1 contributes much more than X_2 to Y, which is logical as if the variance is really small for one measuring system we should trust it as much as possible.

C) The variance of Y becomes:

$$Var\{Y\} = Var\{\frac{\sigma_2^2}{\sigma_1^2+\sigma_2^2}X_1 + \frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2}X_2\} = \left(\frac{\sigma_2^2}{\sigma_1^2+\sigma_2^2}\right)^2 Var\{X_1\} + \left(\frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2}\right)^2 Var\{X_2\} =$$

$$\left(\frac{\sigma_2^2}{\sigma_1^2+\sigma_2^2}\right)^2 \sigma_1^2 + \left(\frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2}\right)^2 \sigma_2^2 = \frac{(\sigma_2^2)^2\sigma_1^2 + (\sigma_1^2)^2\sigma_2^2}{(\sigma_1^2+\sigma_2^2)^2} = \ldots = \frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}$$

From the expression it can be seen that fusing two independent measurements always gives a smaller variance than the smallest variance of X_1 and X_2.

D) If the variance for some reason doesn't represents the error (e.g. there is some kind of bias in the measurements) that means to much (or not enough) of this value contributes to Y.

1.3) One way of calculating the uncertainty of $\hat{\varphi}$ is to use the law of error propagation, i.e. calculate the Jacobian matrices with respect to the uncertain parameters, which gives:

$$\sigma_{\hat{\varphi}}^2 = \left(\frac{\partial\hat{\varphi}}{\partial x_R, \partial y_R, \partial\theta_R}\right) P_{x_R, y_R, \theta_R} \left(\frac{\partial\hat{\varphi}}{\partial x_R, \partial y_R, \partial\theta_R}\right)^T + \left(\frac{\partial\hat{\varphi}}{\partial x_B, \partial y_B}\right) P_{xB, yB} \left(\frac{\partial\hat{\varphi}}{\partial x_B, \partial y_B}\right)^T$$

1.4) A) In the prediction step, the <u>reference model is transformed</u> to the state where the next observation is taken, i.e. predicting what we expect to see (what we expect to observe) in the next state. The <u>uncertainties of the expected observations are predicted</u> at the same time. The uncertainties of the predicted observations are <u>based on e.g. the uncertainty of the robots position, the uncertainty of the beacon locations, the transformation used when predicting the observations</u> etc.

B) In the matching step, the <u>predictions are compared to the observations</u>. To avoid wrong matching of features the matching step should include some kind of <u>decision making on whether an observed feature corresponds to the predicted one</u> or not. This decision is often based on the uncertainty of e.g. the robots position

C) In the update step, the <u>difference between the predicted and observed state is used to update the current state</u> (e.g. the estimate of the robots position). The uncertainty of the observation and the uncertainty of the prediction determine how much the current state is corrected (updated).

1.5) A) The purpose of the validation gate is to determine if an observation corresponds to a prediction, i.e. if we really sees what we have expected.

B) If the validation gate is too small (too narrow) that means many observations are wrongly thrown away as they differ too much from the predictions.

C) If the validation gate is too large (too wide) that means that many of the observations fits to several references, i.e. the system will have ambiguity problems

1.6) A) With Position Probability Grids, the entire state space (all possible places where the robot can be in) are divided into discrete states, referred to as cells. Each cell represents one possible location (x, y θ) and the probability of the robot being in that location.

B) Three problems with the Markov localization techniques could be; long execution time, low accuracy, memory consuming. 1) The long execution time is caused by a huge state space, which takes long time to update after the robot has either moved or sensed the environment. The particle filter method represents the entire distribution by a number of samples drawn from it, which means lesser states to update. 2) The low accuracy is caused by the discrete nature of the cells as the cell size determines the accuracy. The particles can represent arbitrary location and by that perfect accuracy. 3) The huge amount of cells contributes a lot to the large memory consumption.

As the particle filter doesn't keep the entire distribution (all possible locations) but instead represents the distribution by a smaller number of samples means that the particle filter method use lesser memory.

Part II: Solutions to Deeper understanding of a Scientific Paper

The following questions are all on the attached paper, 'Blanche – An Experiment in Guidance and Navigation of an Autonomous Robot Vehicle' by Ingemar Cox, 1991.

A. *The entire equation describes how a laser measurement is transformed from relative co-ordinates (r: range measured by the sensor, α: direction of the measurement given in the sensor co-ordinate system) to global co-ordinates (x: global co-ordinate, y: global co-ordinate). The values; x, y, r and α are scalars. R is a matrix that describes the position of the sensor relative the cart (vehicle rotational centre) and C is a matrix that describes the position of the vehicle relative the world co-ordinate system (base co-ordinate system).*

B. *The purpose of the Cox scan matching algorithm is to compare observations (a set of range readings) to a set of line segments (which are often referred to as the reference map) by translating and rotating the set of range readings. For this the algorithm consists of three steps, targeting, minimization and transformation. The steps are iterated until the process converges, which usually takes somewhere between 5 and 10 iterations. The targeting step does, for each point, first a transform of the range readings into global co-ordinates and then finds the line segment that has the shortest distance to the point. The minimization step finds the congruence (rotation followed by a translation) that minimizes the sum of squared distances between points and their target lines. The congruence is found by applying the same rotation and translation to the entire set of range readings. It is possible to form a set of linear equations by using the pseudo-rotation of the data set. Finally the transformation step translates and rotates the set of range readings by the found congruence and adds the congruence to the total congruence. After this the process is repeated. The error minimized is the distance from a point perpendicular to its target line.*

C. *The process stops when the found congruence is small enough.*

D. *An outlier is a data point that does not have anything in common with the rest of the data set, i.e. a range reading that comes from an object not modeled in the reference map. Because the algorithm uses linear regression, and by that minimizes a sum of squared distances, makes it very sensitive to outliers, as even a single outlier can cause the algorithm to fail miserable, i.e. deliver totally wrong results. One way of rejecting outliers is, in the targeting step, to reject such points that have distances longer than a predetermined threshold value. This threshold can be based on the uncertainty of e.g. the robots position.*

E. *As the co-variance matrix should represent the error of the matching result it should show a large variance in those directions (x, y, θ) in which it makes large errors. In the corridor case (with the ends not visible to the sensor) it means that the algorithm can deliver good results in x and θ but not in y why the variances should be small for x and θ and infinite for y. (Because the sensor data is taken exactly in the middle of the corridor and the robot is rotated in the same way as the world co-ordinate system the covariance of any two parameters are uncorrelated, i.e. equal to 0. This is also the case for the other two examples, i.e. the parameters are uncorrelated.) The square environment should result in small variances for all three parameters. The circular environment (and standing in the midlle of it) should result in small variances for x and y but infinite for θ because it is not possible to determine the rotation.*