Computer Science 226

Grading Criteria for Assignment #2
(Winter 2006)

January 22, 2006

NOTE: Grading criteria are preliminary and given here to illustrate the requirement of your bicycle motion model.

1. Consider a simple kinematic model of an idealized bicycle. Both tires are of diameter d, and are mounted to a frame of length l. The front tire can swivel around a vertical axis, and its steering angle will be demoted α. The rear tire is always parallel to the bicycle frame and cannot swivel.

   For the sake of this exercise, the pose of the bicycle shall be defined through three variables: the x-y location of the center of the front tire, and the angular orientation θ (yaw) of the bicycle frame relative to an external coordinate frame. The controls are the forward velocity v of the bicycle, and the steering angle α, which we will assume to be constant during each prediction cycle.

   Provide the mathematical prediction model for a time interval Δt, assuming that it is subject to Gaussian noise in the steering angle α and the forward velocity v. The model will have to predict the posterior of the bicycle state after Δt time, starting from a known state. If you cannot find an exact model, approximate it, and explain your approximations.

   Grading ✓ Formulation of a motion model. Example: The path of my bike is modelled as an arc on a circle of radius r centred at (x_c, y_c) where (x_c, y_c) is given by $f(x_t, y_t, l, \theta_t, \alpha_t)$. The distance travelled along the arc during an time interval Δt is given by $g(d, v, \Delta t)$, etc.

   ✓✓✓ Computation of posterior distribution. See textbook Table 5.1. You should demonstrate that you can take a robot motion model and derive its posterior probability distribution. Since the textbook already derived the motion model for a garbage-can robot, you should choose a *different* motion model–in particular, one that takes into the account of effects due to the rear wheel.

2. Consider the kinematic bicycle model from Question #1. Implement a sampling function for posterior poses of the bicycles under the same noise assumptions.

   For your simulation, you might assume $l = 100cm$, $d = 80cm$, $\Delta t = 1sec$, $|\alpha| \leq 89^\circ$, $v \in [0; 100]cm/sec$. Assume further that the variance of the steering angle is $\sigma_\alpha^2 = 25^{\circ 2}$ and the variance of the velocity is $\sigma_v^2 = 50cm^2/sec^2 \cdot |v|$. Notice that the variance of the velocity depends on the commanded velocity.

For a bicycle starting at the origin, plot the resulting sample sets for the following values of the control parameters:

problem number	α	v
1	$25°$	$20cm/sec$
2	$-25°$	$20cm/sec$
3	$25°$	$90cm/sec$
4	$80°$	$10cm/sec$
1	$85°$	$90cm/sec$

All your plots should show coordinate axes with units.

Grading ✓ Sample from the posterior distribution. See textbook Table 5.3.
✓ Results from the five trials.

3. Consider once again the kinematic bicycle model from Question #1. Given an initial state x, y, θ and a final x' and y' (but no final θ'), provide a mathematical formula for determining the most likely values of α, v, and θ'. If you cannot find a closed form solution, you could instead give a technique for approximating the desired values.

   Grading ✓✓ Compute $\arg\max_{\theta_{t+1},u,\alpha} P(\theta_{t+1}, u, \alpha | x_t, y_t, \theta_t, x_{t+1}, y_{t+1})$

4. As noted in the text, Monte Carlo localization is *biased* for any finite sample size—i.e., the expected value of the location computed by the algorithm differs from the true expected value. In this question, you are asked to quantify this bias.

   To simplify, consider a world with four possible robot locations: $X = \{x_1, x_2, x_3, x_4\}$. Initially, we draw $N \geq 1$ samples uniformly from among those locations. As usual, it is perfectly acceptable if more than one sample is generated for any of the locations X. Let Z be a Boolean sensor variable characterized by the following conditional probabilities:

$$
\begin{array}{rclcrcl}
p(z \mid x_1) & = & 0.8 & \qquad & p(\neg z \mid x_1) & = & 0.2 \\
p(z \mid x_2) & = & 0.4 & & p(\neg z \mid x_2) & = & 0.6 \\
p(z \mid x_3) & = & 0.1 & & p(\neg z \mid x_3) & = & 0.9 \\
p(z \mid x_4) & = & 0.1 & & p(\neg z \mid x_4) & = & 0.9
\end{array}
$$

   MCL uses these probabilities to generate particle weights, which are subsequently normalized and used in the resampling process. For simplicity, let us assume we only generate one new sample in the resampling process, regardless of N. This sample might correspond to any of the four locations in X. Thus, the sampling process defines a probability distribution over X.

   (a) What is the resulting probability distribution over X for this new sample? Answer this question separately for $N = 1, \ldots, 10$, and for $N = \infty$.

   Grading ✓ Correct distribution for $N = 1$.
   ✓ Correct distributions for $N = 2, \ldots, 10$.

✓ Correct distribution for $N = \infty$.

(b) The difference between two probability distributions p and q can be measured by the KL divergence, which is defined as

$$KL(p, q) = \sum_i p(x_i) \log \frac{p(x_i)}{q(x_i)}$$

What are the KL divergences between the distributions in (a) and the true posterior?

Grading ✓ Computation of KL-divergence

(c) What modification of the problem formulation (not the algorithm!) would guarantee that the specific estimator above is unbiased even for finite values of N? Provide at least two such modifications (each of which should be sufficient).

Grading ✓ Modification #1

✓ Modification #2