WORKSHEET

U7.3 Moving the cloud

Localisation · University · about 30 min

Name _____ Class _____ Date _____

What this lesson is about

The motion update: every particle drives, and every particle's own error goes with it.

- How much noise to add
- This is where the model goes

Questions

6 marks in all

1. A motion update moves every particle by the measured motion with no random term. What goes wrong? [1 mark]

- ☐ **A** The cloud keeps its shape for ever, so the filter claims to be as certain after ten metres as at the start
- ☐ **B** The cloud spreads without limit and the estimate becomes vague
- ☐ **C** The particles drift away from the robot's true motion
- ☐ **D** The weights stop summing to one

2. The noise added per step is much smaller than the robot's real motion error. What is the likely result? [1 mark]

- ☐ **A** An over-confident cloud that shrinks onto a wrong answer and refuses to be corrected
- ☐ **B** A vague estimate because the cloud spreads faster than readings can pull it in
- ☐ **C** Exactly the same estimate, since the measurement update corrects it every tick
- ☐ **D** The effective sample size rises towards N

3. Each particle gets independent noise with standard deviation 0.35 cm per step. After 40 steps, what spread (standard deviation) should the cloud have, in cm to two decimal places? [1 mark]

_____

4. You double the per-step noise standard deviation. What happens to the spread of the cloud after 5 seconds? [1 mark]

- ☐ **A** It doubles
- ☐ **B** It quadruples
- ☐ **C** It grows by a factor of sqrt(2)
- ☐ **D** It stays the same, because it depends only on the number of steps

5. A Kalman filter for the same robot would use a process noise Q of 0.16 cm squared per step. What standard deviation, in cm, is a reasonable starting point for the per-particle motion noise? [1 mark]

_____

6. Which of these belong in the motion update of a particle filter? [1 mark]

Tick every answer that is true.

- ☐ **A** Speed proportional to the command, with a dead band
- ☐ **B** More error when turning than when driving straight
- ☐ **C** Removing any particle that would pass through a wall
- ☐ **D** Weighting each particle by how well it predicts the distance reading

The task: move the cloud

Start every particle at zero, move them with the flow sensor plus their own noise as the robot drives at least 40 cm, plot `mean y` and `spread y`, and print both at the end.

```
from bugbot import *
import random
connect()

DT, N = 0.1, 300
ys = [0.0] * N
```

Plan your program here, then type it in and press Run.

Do it on the robot

www.bugbotlab.com/learn/u7-3-moving-the-cloud/

The simulator checks it and tells you when it passes. Nothing to install, no account.

Challenges

1. Run it with no random term at all. What does the spread do, and why is that a lie?
2. Double the noise and compare the spread at the end. Does it double?
3. Start every particle at 40, where the robot starts on the mat, so the cloud is in mat coordinates. Drive on towards the far wall and kill any particle that goes past 200. How many do you lose, and when?