



U3.3 How the error grows

Name _____

Class _____

Date _____

What this lesson is about

Noise walks, bias marches, scale stretches, and a heading error rotates everything after it.

- Noise: a random walk
- Bias: a march
- Scale: a stretch
- Heading error: the multiplier
- The ranking

Questions

6 marks in all

1. The noise error in a dead reckoning run grows as a random walk. If the run is made twice as long, by what factor does that error grow? Give two decimal places. [1 mark]

2. Which of these errors grow in direct proportion to the time or distance driven, rather than with its square root? [1 mark]

Tick every answer that is true.

- ☐ A Heading error from a gyro bias
- ☐ B Position error from a flow scale error
- ☐ C Error from zero-mean noise on flow()

3. Gyro rates in deg/s are read while the robot stands perfectly still. What does this print? [1 mark]

```
rates = [0.3, 0.5, 0.2, 0.4, 0.1, 0.3]
mean = sum(rates) / len(rates)
spread = (sum((r - mean) ** 2 for r in rates) / len(rates)) ** 0.5
print("bias:", round(mean, 2))
print("noise:", round(spread, 2))
print("after 60 s:", round(mean * 60, 1))
```

4. Using the lesson's rule that driving d with a heading error of e radians puts you about $d \times e$ to one side, this works out the sideways error over 10 metres. What does it print? [1 mark]

```
import math
for deg in (1, 3, 10):
    print(deg, round(1000 * math.radians(deg), 1))
```

5. The flow sensor reads 4 percent high. How many centimetres too far does dead reckoning put the robot [1 mark] after a 10 m straight run?
-
6. For this sort of ground robot over a run of a minute or two, which error source does the lesson rank as [1 mark] the most damaging?
- ☐ A Heading bias
 - ☐ B Scale error on the translation
 - ☐ C Slip of the drive on the mat
 - ☐ D Noise on the readings

The task: measure the gyro bias

Without moving the robot at all, print this gyro's bias as `bias: 0.31`, in degrees per second.

```
from bugbot import *
connect()

rates = []
```

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



Do it on the robot

www.bugbotlab.com/learn/u3-3-how-error-grows/

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

Challenges

1. Measure the bias five times. Is it the same number each time? Is the noise?
2. Run `set_noise(3)` and measure again. Which changes, the bias or the spread?
3. Work out how long the robot can dead reckon before the bias alone costs it 10 cm, at 20 cm/s.