COMSM4111 Robotic Systems

CW1 Simulated Robot Localisation

Individual Assignment

2 Courseworks that add to 100%

- **50% Individual component**
  - Develop Particle Filter Localisation weeks 2-6.
  - MATLAB code submission using BotSim.
  - Deadline via SAFE: 6th Mar 2018, 5pm
  - Live marking session 7th Mar 2018
- **50% Team work**
  - Develop solution for kidnapped-robot problem, weeks 7-11.
  - MATLAB code deadline via SAFE: 24th Apr 2018, 5pm
  - Robot marking session in Lab: 25th Apr 2018

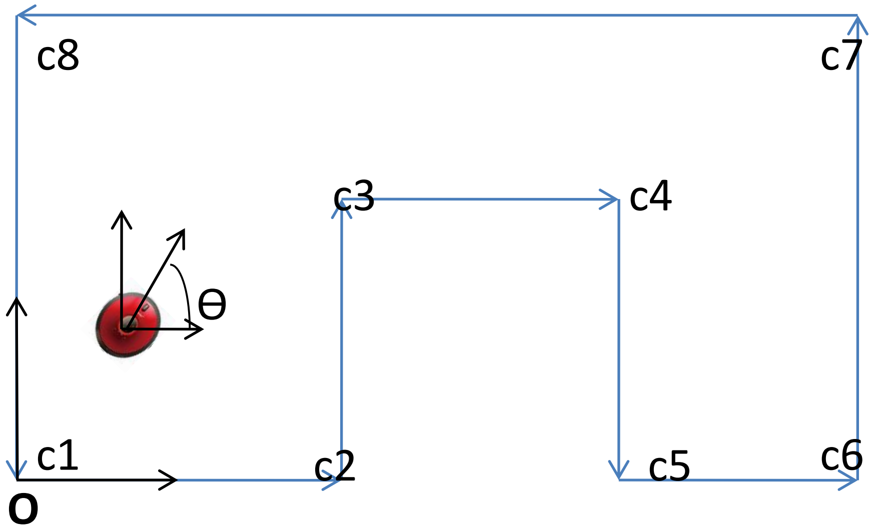
Task

- Develop a **localisation system** that takes the robot from a random initial place in the arena to a specific target location.

- I will give you a vector of **map** vertices, a **robot starting** position (required for the simulator only) and **ONE target** point. Your simulator should give me as output the final robot **estimate in x,y**. You **must use the BotSim** output format.

Kidnapped-robot localisation task

- Robot will be placed in a **random location**.
- Robot has to **identify where it is** and go as quick as possible to a target point inside arena.



Map and targets will be given as a Matlab vector of coordinates

e.g. M=[0 0;20 0;20 15;40 15;40 20;0 20]

T=[5,15], S=[40,85]

Getting started...

- **BotSim library (for CW1/2)**is our own library for the unit. Download it from Blackboard. Follow the examples and the PDF guide.

- **Matlab Toolbox (for CW2)** is the free RWTHMindstormsNXT toolbox from Aachen University:

  http://www.mindstorms.rwth-aachen.de/trac/wiki/Download

- You may install these in your personal machines **but marking will be done using the lab machines and what is installed there, in the interest of fairness.**

Getting started....

- Matlab is installed on all windows' lab machines BUT the **RWTH - Mindstorms NXT Toolbox** may only be installed *in labs 1.07 and 1.08* in MVB. Feel free to try in other windows labs in the Faculty.

- On the lab's machines the directory for e.g. toolbox demos is (but please check may have changed):

```
C:/Program' Files'/MaTLAB/R2010a/toolbox/RWTHMindstormsNXT/demos/
```

First Assignment (CW1)

- Develop a localisation system for a simulated 2D "point" robot (x,y,theta).

↳ ie no padding on walls etc.

Hint: use a particle filter as discussed in lectures, but you can propose a different strategy too.

I Strongly suggest you talk to me to discuss a strategy which is different from a particle filter *before* you implement it

Object oriented programming
↳ look up.

BotSim

- It does actually many of the **things for you** but clearly not all. Please **use the examples** to start learning about it.

- BotSim was authored by one of our PhD Students, Austin Gregg-Smith. It uses object oriented programming so elements like the particles are software objects.

- Fresh updated version every year. We are always open for improvement suggestions

- Now, let me show you some **Demo Examples …**

CW1 Simulator (max 100 points):

Simulator's features	(up to 20 points)
Simulated Path planning	(up to 30 points)
Time-To-Complete	(up to 10 points)
Localisation accuracy	(up to 40 points)

CW1 Simulator Details

- **Simulator's features (up to 20 points):**
  - Robustness of navigation (no collisions), efficiency of algorithm (short path length) as discussed in Example 6 of BotSim.
  - Your algorithm should deal with local map symmetries

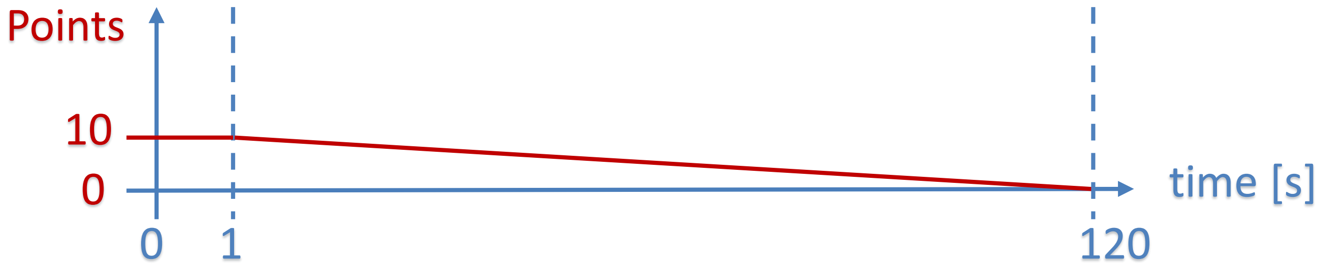
Look up

  - Your code should be tidy and safe e.g. in terms of potential infinite loops, exception handling, don't use NaN/Inf conditions. You must not use TIC … TOC in submitted code (ok for testing)
- **Simulated Path planning (up to 30 points):**
  - Demonstrate your simulated robot can create an efficient (i.e. short) path to the target
  - The robot navigates along the path in a robust way e.g. in case of wrong localisation, the robot should not try to follow a path through walls
  - We expect a plotting option which might be disabled during marking to save time

CW1 Simulator Details

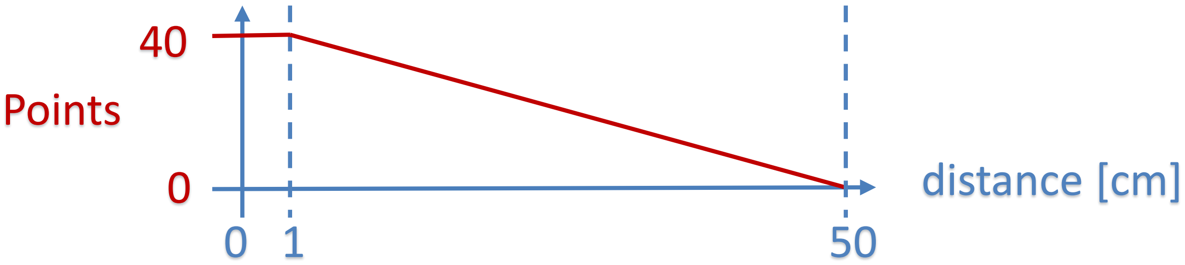
- **Time-To-Complete (up to 10 points)**

  The less time you need, the higher your rating



- **Localisation accuracy (up to 40 points)**

  The close you get to the target, the higher your rating



Points are summed and rounded (0.5 and bigger = 1, else 0)

CW1 Simulator Details

- If your system can do path planning only, please indicate that at the top of the localise.m file and **contact a member of the teaching team** during the live marking session.
- In that case, the path plotting has to be enabled

Submission

- Submit all files required to run your solution e.g. localise.m plus path planning functions
- Except, don't upload BotSim (we will aske you to download a fresh copy for assessment anyway)
- Combine all files in a ZIP-file before uploading if you have more than 2 files

Simulator is tested *exclusively* from what you submitted online

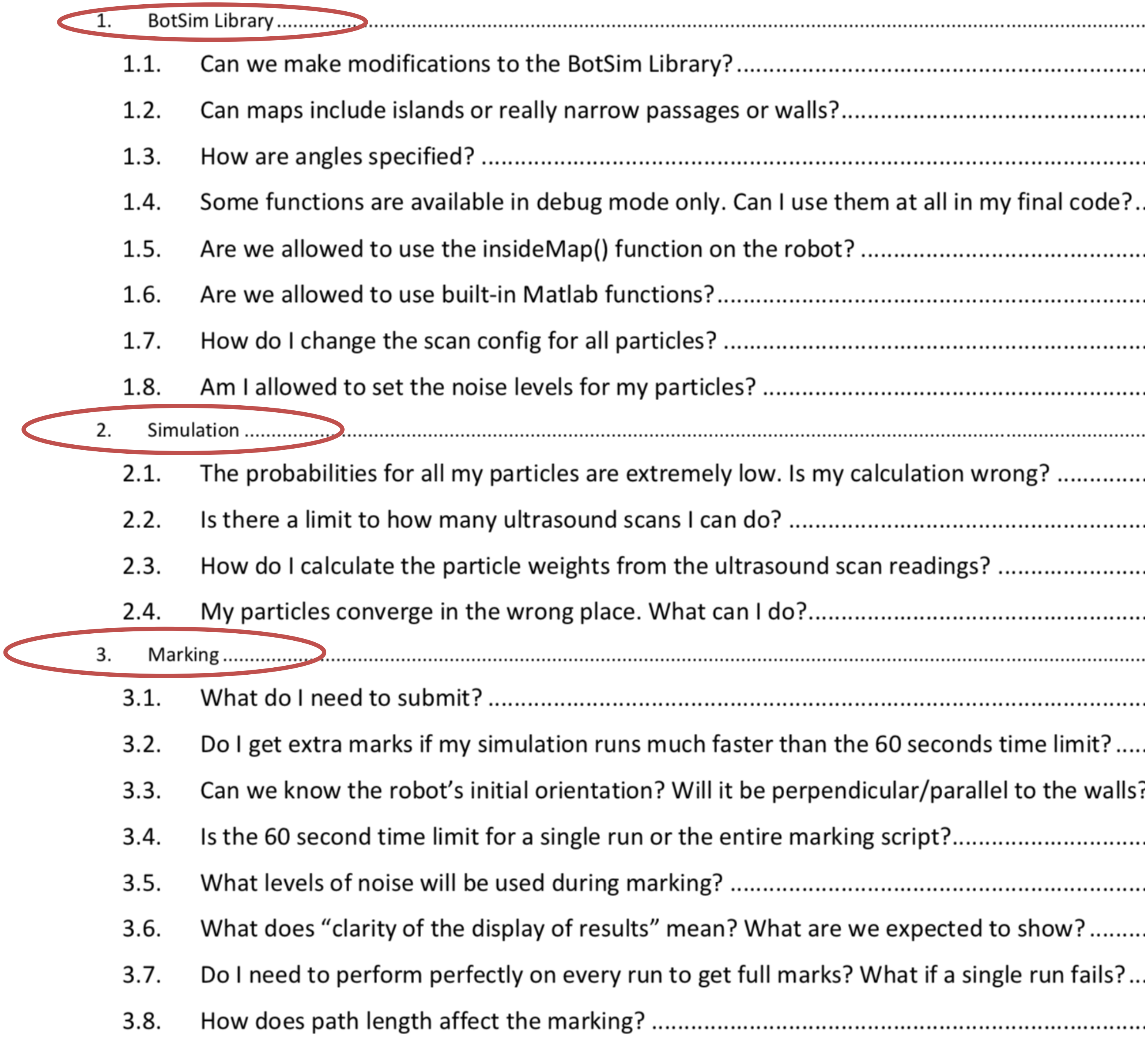
TO GET POINTS YOUR SIMULATOR MUST USE THE BOTSIM Library Format for the

Auto Marking, see respective example.

Getting help

- We have **labs** for you to ask questions live to me and the lab teaching assistants. Thursdays.
- We have setup a **forum in Blackboard** for you to post questions. This is my preferred way to dealing with questions so others benefit from answers plus 100 people emailing me would not work(!).
- You are encouraged to **discuss things with your team** members (albeit this CW1 is individual so code sharing will trigger plagiarism penalty). Or with others in the class.
- Find answers to your questions in the **FAQ**s

FAQ CW1



Summary

- You can use your own localisation method though perhaps **particle filters** is a good place to start.
- **Familiarise yourself with BotSim** and the RWTH toolbox.
- Likely there will be many questions, use the relevant **discussion in Blackboard** to post questions, I will monitor it. Have a look at the **FAQs** and use the **Lab** sessions to ask for feedback.

Resources for learning Matlab

Good starting point and tuned to Engineering/Science students:

- www.mccormick.northwestern.edu/docs/efirst/matlab.pdf

Yet another introduction:

- www.mathworks.com/moler/intro.pdf

BUT: Before you delve into Matlab or Robot hardware, try to think first what is your overall strategy, then implement.