AMME5520: Project Part 1

Ian R. Manchester

Due date: Sunday 10th of April, 11:59pm (end of wk 7)

The following questions explore motion planning via the material from weeks 1-3. Submit a report through TurnItIn justifying the design decisions you have made and exploring their consequences. The report must be thorough but concise (feedback from your tutor will be valuable in achieving the right balance).

All matlab code must also be submitted through TurnItIn. While you are encouraged to discuss approaches with tutors and your fellow students, all code, analysis and submitted writing must be entirely your own work.

Problem Specification

As part of its collaboration with the NASA Artemis mission[1], Australia will be designing a lunar rover for regolith collection, with the aim of potentially supplying oxygen to a lunar base.

You are working as a navigation and control engineer as part of the mission, and this project is to develop some early-stage design concepts, simulations, and evaluations that will inform the final design.

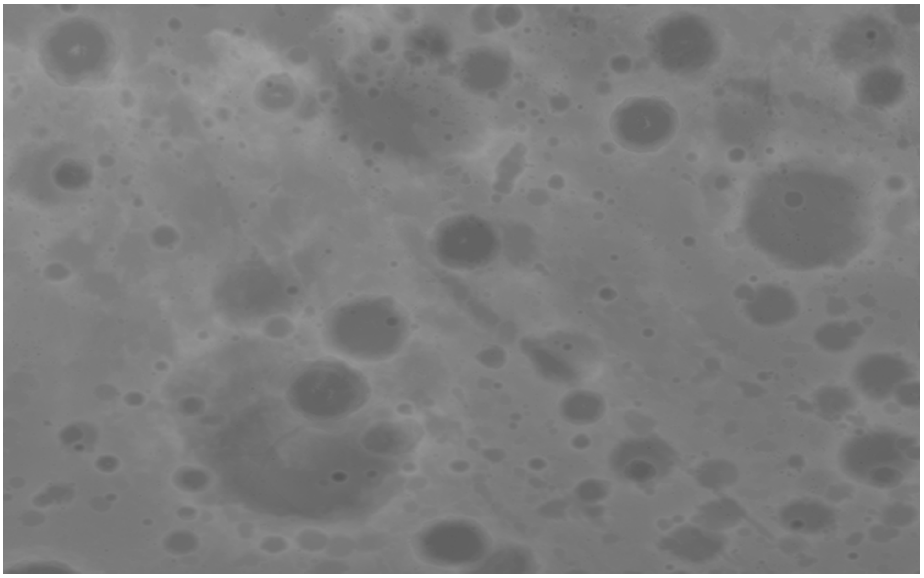
Problem 1 [50 marks]

NASA and JAXA have created elevation maps of the surface of the moon based on a combination of laser and stereo-camera imagery[2].

An example is shown below, where darker areas are low-lying areas, and lighter areas are raised areas. You can see many craters of different sizes.

[1] In Greek myth, Artemis was the twin sister of Apollo.

[2] Barker et al, A new lunar digital elevation model from the Lunar Orbiter Laser Altimeter and SELENE Terrain Camera, *Icarus*, 2016



You can download this image from canvas (`moon.png`) and load it into Matlab with the `imload` command, and display it with `imshow`.

Your task is to design and analyse a system for path planning from any given point to any other point in this map, by traversing as much as possible only flat regions to minimize risk of losing the rover. The following workflow and discussion points for your report are suggested:

1. (5 marks) Estimate a traversability map of the terrain by measuring its local deviation from flatness. You might want to read the documentation for the `stdfilt` command.

2. (20 marks) Implement Dijkstra's algorithm to find shortest paths between points on this map, and test it with a variety of initial and final locations.

3. (10 marks) Visualise the arrival cost from a particular location, similarly to the global travel map in Lecture 3. Discuss the results.

4. (15 marks) Propose a heuristic and implement $A^\star$. Is your heuristic admissable and consistent? What happens if you use a heuristic which isn't? Compare to Dijkstra's algorithm in terms of the final result and computational effort for a variety of start and end goals.

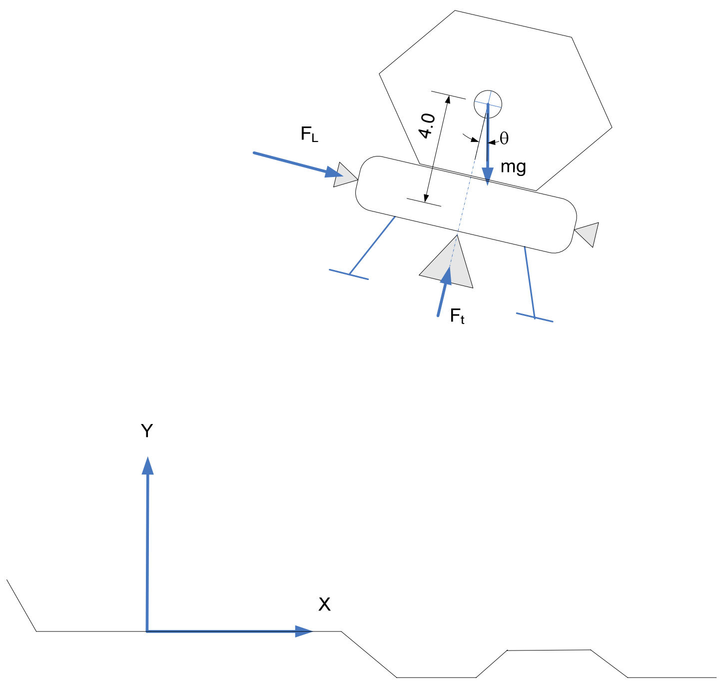
For this question you may use only basic Matlab functions and the functions described above from the Image Processing toolbox. In particular, you may **not** use any functions or toolboxes that implement graph-search algorithms such as Dijkstra or $A^\star$. You may, however, use the Matlab Grader problem from Week 3 as inspiration.

Basic Matlab functions are considered to be those found in the documentation (type `doc` in the command window) under the following headings: Language fundamentals, Graphics, Programming Scripts and Functions, and under Mathematics the following subheadings: Elementary Math, and Linear Algebra.

Problem 2 [40 marks]

Since you did such a great job on the rover path planning, you've been asked to help out the team designing control system for the the lunar lander.

A simple 2D schematic is shown below:



Some key parameters are collected in the table below:

Specification	Value
Mass	20,000kg
Moment of Inertia	100,000 kg m^2
Max. F_t propulsion	44kN
Max. F_L propulsion	0.5kN
Gravitational Constant	1.6ms^{-2}

The following tasks are suggested:

1. (10 marks) Construct a nonlinear state-space model of the dynamics of the lander describing its translational and rotational dynamics.

2. (10 marks) Linearise the dynamics about a steady hover state producing a model of the form
$$\dot{x} = Ax + Bu$$

3. (20 marks) Assuming all states are measurable, design an LQR controller to stabilise the hover state, and simulate it with the original nonlinear dynamics.

   - Examine the effect of different choices of weighting matrices.
   - Examine the response to different initial conditions.
   - Examine the similarity/differences between the linearised and nonlinear models.

Report Quality [10 marks]

In addition to the marks above for technical work, 10 marks will be allocated for the presentation and structure of your report and any associated code.

It is expected that your report is well-written, clearly formatted, design decisions are well justified, and the results are concisely but thoroughly analysed, and that any Matlab code is well-structured, legible, and properly commented. Any figures should be legible and clear.

You may provide short code snippets within the report to illustrate ideas, and you must also upload all code