Module Code
[COM00052H]

BEng, BSc, MEng, MMath and MSc Degrees 2023–24

Open Assessment

Department: Computer Science

Module: Autonomous Robotic Systems Engineering (AURO)

Title: Individual coursework

Issued: Thursday 09/11/2023

Submission due: 12 noon, Thursday 18/01/2024

Feedback and Marks due: Thursday 22/02/2024

Instructions:

All students should submit their answers through the electronic submission system: the Department's Teaching Portal by 12 noon, Thursday 18/01/2024. An assessment submitted after this deadline will be marked initially as if it had been handed in on time, but the Board of Examiners will normally apply a lateness penalty.

Your attention is drawn to the section about Academic Misconduct in your Departmental Handbook.

Any queries you may have on this assessment should be posted on the Discussion Board on the Virtual Learning Environment (VLE) page for Autonomous Robotic Systems Engineering (AURO) in the appropriate discussion area. **No questions will be answered after 11/01/2024**.

1 Task definition

This assessment focuses on the retrieval of items by autonomous mobile robot(s). You will be provided with a simulated world containing items distributed throughout the environment, which the robot(s) must collect and return to a home zone. This simulated world will also contain obstacles that the robot(s) need to avoid.

A robot can collect an item by driving into it, and will 'hold' the item until it has returned to the home zone. A robot cannot hold more than one item at a time. Once an item has been collected and returned home, a new one will automatically spawn to replace it.

You must design and implement a solution to this task, using mobile robot(s) that autonomously collect the items and return them home in an efficient manner.

2 Deliverables

This assessment is worth 100% of the module mark, and is made up of 3 parts with the following weightings:

1. **Implementation:** 40%
2. **Demonstration:** 10%
3. **Report:** 50%

Guidance on what you are required to do for each part of the assessment is provided in the following sub-sections. A full marking rubric can be found in Section 4.

2.1 Implementation (40%)

You should engineer an autonomous robotic system that solves the assessment task efficiently, while obeying the following constraints:

- You must implement your solution in simulation, using the Robot Operating System (ROS). Specifically, you must use ROS 2 Humble Hawksbill, Gazebo Classic 11, and the Python client library `rclpy`.
- You must use autonomously controlled (i.e. not teleoperated) TurtleBot3 Waffle Pi robot(s). Your solution may use up to 3 robot(s), and they must all start in the home zone.
- You must include a README file that describes how to run your code, and details any packages or environment variables that your implementation depends upon.

There are many different ways of approaching the assessment task, so there is scope for a variety of solutions. Your implementation will be assessed against the following criteria:

Sophistication [12 marks]: You will be assessed on the intelligence of your autonomous robotic system, and how efficiently it addresses different aspects of the task. Multi-robot systems may be considered more sophisticated than single-robot systems, but only if they are implemented appropriately.

System architecture [12 marks]: The architecture of your autonomous robotic system will depend upon your approach to the assessment task. You will be assessed on the modularity of your solution – i.e. how well it separates functionality into reusable components with appropriate interfaces between them.

Use of ROS concepts [8 marks]: Your solution should use a variety of ROS concepts. You will be assessed on the breadth of the ROS features that your implementation uses, and how well you have demonstrated your understanding of them (through correct usage).

Understandability [8 marks]: Your implementation should be easy to understand. You will be assessed on the structure of your code, use of sensible naming conventions and comments, and how comprehensive your README file is.

You will be provided with code that defines the assessment task, which should be downloaded from the VLE. This code sets up the simulated world, and runs a ROS node that spawns items and keeps track of when they are collected and returned home by the robot(s). You will also be provided with a ROS node that processes images from a robot's camera, and publishes information about any items that are detected. You must use this code **without modification**, as it defines the assessment parameters of the task.

You are allowed to use any of the example code provided as part of the teaching delivery for the AURO module, including solutions to practical exercises. The use of 3rd party packages is also permitted, as long as you cite the original source and include instructions on how to build them.

2.2 Demonstration (10%)

You must demonstrate your solution by recording a video that showcases its features, as well as its performance with respect to the assessment task.

- You video should be a screen recording that demonstrates the operation of your autonomous robotic solution in both the Gazebo simulator and the RViz visualisation tool.
- You should not show your code (this will be assessed separately), but you should show the terminal commands used to run your solution.
- You must provide captions that explain what is being shown in your video as it progresses.

The video demonstration is worth **[10 marks]**. You will be assessed based on how well the video demonstrates your implementation, and how informative the associated captions are. The purpose of this video is only to demonstrate that your solution performs as described in your report. You do not need to include any discussion, analysis, or evaluation in this video – this should be included in your report instead.

Your video must be no longer than 5 minutes in duration. If your video exceeds this time limit, the marker will stop watching after 5 minutes, and base the mark on what they have seen so far.

2.3 Report (50%)

You must write a report that details the design and implementation of your solution. You should also analyse the performance of your solution, and present the results in your report. Finally, your report should evaluate the strengths and weaknesses of your solution, and reflect on related safety implications and ethical considerations. The report must be structured as follows:

Design [8 marks]: This section should describe the design of your system and justify your design decisions. It should also include a diagram that communicates the high-level design of your system, and how the individual components interact (e.g. a block diagram).

Implementation [7 marks]: This section should describe and justify your implementation, with particular reference to your use of ROS concepts. It should also include a diagram that communicates how your solution achieves autonomy (e.g. a state machine). You should not include your code here, as this will be assessed separately.

Analysis [10 marks]: In this section, you should describe and justify your experimental approach to analysis. You should then present and interpret the results of your analysis, which should be both qualitative and quantitative. You may wish to present your data in the form of figures and/or tables.

Evaluation [10 marks]: This section should include a discussion of the strengths and weaknesses of your solution. Your evaluation should be based on your design, implementation, and analysis. It should also include a discussion of how well your solution would transfer from simulation to reality, and how this could be improved.

Safety and ethics [10 marks]: This section should include a discussion of safety implications and ethical considerations related to item retrieval by autonomous robotic systems in real-world scenarios. You should also reflect on how these topics would relate to the approach taken by your solution, if it were to be implemented in a real-world scenario.

The remaining **[5 marks]** of the report mark will be based on its presentation (structure, figures, adherence to the template, and use of referencing).

Your report must be formatted using the IEEE conference template (either LaTeX or Microsoft Word) [1], and any references must follow the IEEE referencing style[2]. You must use A4 paper size (not US letter, which is the default for the LaTeX template), and you must not edit the formatting (e.g. font, margins, columns).

You should not include an abstract, introduction, literature review, or any keywords. You should only include your examination number in the author field – do not include your name, username, email address, or any other identifying information. You should also include the module code COM00052H on the first page.

Your report must be no longer than 4 pages (excluding references). If your report exceeds this page limit, the marker will stop reading when they reach the limit, and base the mark on what they have read so far.

3 Submission

The submission of your implementation, demonstration, and report for this assessment should abide by the following rules.

3.1 Anonymity

You should identify yourself only by your examination number. **Your name, username, email address, or any other identifying information must not be present anywhere in your submission**.

This includes all ROS package metadata, code comments, environment variables, Linux terminal prompts, PDF metadata, video captions, etc. You must not narrate your video demonstration, nor include any audio or visuals that could deanonymise your submission.

3.2 Electronic submission

You should combine all of the files for your implementation, demonstration, and report into a single ZIP file and submit it via the Teaching Portal. The contents of this ZIP file should be structured as shown in Figure 1:

It is your responsibility to ensure that your implementation runs under the Departmental ROS environment before submission.

[1] https://www.ieee.org/conferences/publishing/templates.html
[2] https://subjectguides.york.ac.uk/referencing-style-guides/ieee

```
submission.zip
├── report.pdf
├── demonstration.mp4
├── captions.srt
├── README.txt
├── rosgraph.png
└── src
    └── Your ROS packages
```

Figure 1: Required structure of electronic submission ZIP file.

The source code of your implementation should be included in a directory called `src` containing your ROS packages. You should not include your entire ROS workspace – i.e. you should exclude the `build`, `install`, and `log` directories created by `colcon`.

Your README file should be in plain text format. You should also include a PNG file exported from `rqt_graph` that shows the ROS graph of your solution.

Your demonstration video must be in MP4 format, and the captions must be in SRT (SubRip) format.

Your report must be a single PDF file.

4 Marking rubric

The marking rubric for each part of the assessment is given below in the following tables:

1. **Implementation (40%):** Table 1 (pages 8 to 9)
2. **Demonstration (10%):** Table 2 (page 9)
3. **Report (50%):** Table 3 (pages 9 to 12)

Table 1: Implementation (40%)

Mark:	0–29%	30–39%	40–49%	50–59%	60–69%	70–79%	80–100%
Sophistication [12 marks]	Single-robot system that fails to address any part of the task	Single-robot system that fails to address some parts of the task	Single-robot system that sufficiently addresses all parts of the task	Single-robot system that efficiently addresses all parts of the task, or multi-robot system that sufficiently addresses all parts of the task	Single-robot system that intelligently addresses all parts of the task, or multi-robot system that efficiently addresses all parts of the task	Multi-robot system that intelligently addresses all parts of the task	Multi-robot system that intelligently and creatively addresses all parts of the task
System architecture [12 marks]	Monolithic architecture with inappropriate use of interfaces	Monolithic architecture with limited use of interfaces	Monolithic architecture with appropriate use of interfaces	Modular architecture with limited use of interfaces	Modular architecture with appropriate use of interfaces	Modular architecture that separates functionality into reusable components, with appropriate use of interfaces	Modular architecture that separates functionality into reusable components, with considered use of interfaces
Use of ROS concepts [8 marks]	Demonstrates a misunderstanding of ROS concepts	Demonstrates a very limited understanding of ROS concepts	Demonstrates a limited understanding of ROS concepts	Demonstrates an understanding of ROS concepts	Demonstrates a good understanding of ROS concepts	Demonstrates a strong understanding of ROS concepts	Demonstrates a comprehensive understanding of ROS concepts
Understandability [8 marks]	Poorly structured code with unsuitable naming and no comments	Poorly structured code with unsuitable naming and limited comments	Sufficiently structured code with suitable naming and limited comments	Sufficiently structured code with suitable naming and comments where appropriate	Sufficiently structured code with descriptive naming and comments where appropriate	Well-structured code with descriptive naming and comments where appropriate	Exceptionally well-structured code with descriptive naming and comments where appropriate

Continued on next page

Table 1: Implementation (40%) – Continued from previous page

Mark:	0–29%	30–39%	40–49%	50–59%	60–69%	70–79%	80–100%
	No README submitted	README provides insufficient instructions on how to run the code	README provides sufficient instructions on how to run the code	README provides sufficient instructions on how to run the code	README provides sufficient instructions on how to run the code	README provides detailed instructions on how to run the code	README provides comprehensive instructions on how to run the code

Table 2: Demonstration (10%)

Mark:	0–29%	30–39%	40–49%	50–59%	60–69%	70–79%	80–100%
Video [10 marks]	No video submitted, or is in the wrong format	Video poorly demonstrates the implementation, and does not include captions	Video sufficiently demonstrates the implementation, with limited captions	Video sufficiently demonstrates the implementation, with descriptive captions	Video demonstrates the implementation well, with informative captions	Video comprehensively demonstrates the implementation, with informative captions	Video comprehensively demonstrates the implementation, with informative and insightful captions

Table 3: Report (50%)

Mark:	0–29%	30–39%	40–49%	50–59%	60–69%	70–79%	80–100%
Design [8 marks]	No description of design, or very little detail given	Limited design that is insufficiently described and justified	Limited design that is sufficiently described and justified	Appropriate design that is sufficiently described and justified	Good design that is well-described and justified	Well-formulated design that is well-described and justified	Excellent design that is well-described and thoroughly justified

Continued on next page

Table 3: Report (50%) – Continued from previous page

Mark:	0–29%	30–39%	40–49%	50–59%	60–69%	70–79%	80–100%
	No diagram	No diagram, or diagram that fails to communicate the design of the system	Diagram partially communicates the design of the system	Diagram adequately communicates the design of the system	Diagram clearly communicates the design of the system	Diagram comprehensively communicates the design of the system	Outstanding diagram that comprehensively communicates the design of the system
Implementation [7 marks]	No description of implementation, or very little detail given	Implementation is insufficiently described and justified, and fails to demonstrate an understanding of ROS	Implementation is sufficiently described and justified, and demonstrates a limited understanding of ROS	Implementation is sufficiently described and justified, and demonstrates an understanding of ROS	Implementation is well-described and justified, and demonstrates a good understanding of ROS	Implementation is well-described and justified, and demonstrates a strong understanding of ROS	Implementation is well-described and thoroughly justified, and demonstrates a comprehensive understanding of ROS
	No diagram	No diagram, or diagram that fails to communicate the implementation of autonomy	Diagram partially communicates the implementation of autonomy	Diagram adequately communicates the implementation of autonomy	Diagram clearly communicates the implementation of autonomy	Diagram comprehensively communicates the implementation of autonomy	Outstanding diagram that comprehensively communicates the implementation of autonomy
Analysis [10 marks]	No description of experimental approach, or very little detail given	Limited experimental approach that is insufficiently described and justified	Limited experimental approach that is sufficiently described and justified	Appropriate experimental approach that is sufficiently described and justified	Good experimental approach that is well-described and justified	Well-reasoned experimental approach that is well-described and justified	Rigorous experimental approach that is well-described and thoroughly justified

Continued on next page

Table 3: Report (50%) – Continued from previous page

Mark:	0–29%	30–39%	40–49%	50–59%	60–69%	70–79%	80–100%
	No analysis, or very little detail given	Limited analysis that is exclusively qualitative	Sufficient analysis that is exclusively qualitative	Sufficient analysis that is both qualitative and quantitative	Good analysis that is both qualitative and quantitative	Insightful analysis that is both qualitative and quantitative	Insightful and thorough analysis that is both qualitative and quantitative
Evaluation [10 marks]	No discussion of strengths and weaknesses, or very limited detail given	Limited discussion of strengths and weaknesses, and does not relate to the solution	Limited discussion of the strengths and weaknesses of the solution	Sufficient discussion of the strengths and weaknesses of the solution	Good discussion of the strengths and weaknesses of the solution	Insightful discussion of the strengths and weaknesses of the solution	Insightful and thorough discussion of the strengths and weaknesses of the solution
	No discussion of how well the solution would transfer from simulation to reality	Demonstrates a lack of understanding of how well the solution would transfer from simulation to reality	Demonstrates a limited understanding of how well the solution would transfer from simulation to reality	Demonstrates an understanding of how well the solution would transfer from simulation to reality	Demonstrates a good understanding of how well the solution would transfer from simulation to reality	Demonstrates a detailed understanding of how well the solution would transfer from simulation to reality	Demonstrates a comprehensive understanding of how well the solution would transfer from simulation to reality
Safety and ethics [10 marks]	No discussion of safety or ethics, or very limited detail given	Limited discussion of safety or ethics that does not relate these topics to the assessment	Limited discussion of safety or ethics that begins to relate these topics to the assessment	Discussion of both safety and ethics that relates these topics to the assessment	Good discussion of both safety and ethics that relates these topics to the assessment	Insightful discussion of both safety and ethics that relates these topics to the assessment	Insightful and thorough discussion of safety and ethics that relates these topics to the assessment
Presentation [5 marks]	Template not used	Template used with major deviations	Template used with moderate deviations	Template used with moderate deviations	Template used with minor deviations	Template used with very minor deviations	Template used with no deviations

Continued on next page

Table 3: Report (50%) – Continued from previous page

Mark:	**0–29%**	**30–39%**	**40–49%**	**50–59%**	**60–69%**	**70–79%**	**80–100%**
	No structure, or very disjointed	Poorly structured and disjointed	Sufficiently structured, but disjointed	Appropriately structured, with mostly logical flow	Good structure, with logical flow	Well-structured, with clear and logical flow	Well-structured, with clear narrative and logical flow
	No figures, or illegible figures	Figures are illegible or of poor quality	Figures are legible and of acceptable quality, but may not include captions	Figures are of good quality, but may not include captions	Figures are of good quality and include captions	Figures are of high quality and clearly captioned	Figures are of excellent quality and descriptively captioned
	Any references are incorrectly formatted, or not cited	References are incorrectly formatted and not cited	Some references are incorrectly formatted or not cited	Some references are incorrectly formatted or incorrectly cited	Most references are correctly formatted and correctly cited	Most references are correctly formatted and correctly cited	All references are correctly formatted and correctly cited

END OF PAPER