Comparison of SLAM algorithms on virtual test bed URSSA for space applications

Midhun S. Menon[1] and Michael E. Walker[2] and Daniel Koris[2] and Daniel Szafir[2,3] and Jack Burns[1]

***Abstract*— Robots are critical enablers for space exploration as they help offset safety risks associated with a human astronaut, aid in precursor missions prior to manned missions, provide critical on-mission and post-hoc support. However, designing robotic systems to navigate, map and localize itself in hostile and unknown extraterrestrial worlds remains open challenge. The prime reason for this being the testing the semi-autonomous agents for robust performance in such environments. In this paper, we address this issue by using Unity-ROS(Robot Operating System) Simulator for Space Applications(URSSA) for compare performance of three popular Visual Intertial Odometry (VIO) Simultaneous Localization And Mapping (SLAM) algorithms on lunar surface. The test architecture, agent modelling, rationale for test cases and generation of ground truth data is discussed. Paper concludes with results from simulations comparing the algorithms and discusses on failure reasons and potential directions for improving algorithms for better applicability such environments. We believe such a comparative study can give vital pointers for future research directions and help accelerate development of specific algorithms for SLAM in extra terrestrial environments.**

I. INTRODUCTION

The last decade has seen major space faring nations chart out focussed plans for deep space exploration starting with the moon. The Artemis I**(Ref.)**, Chang'e-4 mission**(Ref.)** and Chandrayaan-2**(Ref.)** are some of the ongoing and completed precursor missions to put human explorers on lunar surface. These missions have robotic systems onboard to do remote investigation,survey and data collection of the lunar surface. The follow-up missions to all of these missions have a increasing component of robotic systems of which ground based mobility systems are central. Wheeled rovers are the most commonly used platform for planetary surface mobility. To date, they have been used for surface exploration, mapping and scientific investigation. However, future missions call for advanced mission operations including periodic monitoring of geo-spatially distributed scientific assets, remote assembly and In-Situ Resource Utilization(ISRU)**(Ref.)** tasks. The first of these call for navigation, mapping and localization capabilities over large ranges and the ISRU tasks require advanced manipulation and dexterity capabilities. More importantly, these algorithms need to be tested under a rich and varying environment to make sure they perform reliably. In this paper, the focus is on the testing of algorithms for navigation, mapping and localization. Currently, space robotics community relies on mission analogs for evaluation and testing of such on-board algorithms. These are operations carried out on Earth in a mock-up of the true mission environment. However, mission analogs are incapable of simulating certain critical, unique and finer aspects of the extraplanetary environment (e.g., gravity, photometric anomalies etc.) which impacts testing aefficacy. As an example, visual artefacts present in target environment but not simulated in a mission analog may artificially inflate expectations and over-estimate algorithm performance during testing. On similar lines, locating a matching topography for a mission analog on earth can also prove increasingly challenging as the geological processes governing surface formation and evolution will differ entirely in most cases.

Through this paper, we aim to address the problem of designing virtual test beds to robustly evaluate and design Visual Intertial Odometry (VIO)**(Ref.)** Simultaneous Localization And Mapping (SLAM)**(Ref.)** algorithms specific to extra-terrestrial environments. A case-study of using in-house developed Unity-ROS(Robot Operating System) Simulator for Space Applications(URSSA)**(Ref.)** to test and compare performance of three popular SLAM algorithms in a virtual lunar environment is presented.

Though alternate frameworks exist for simulating the Moon (e.g., [1]), they have not been designed for evaluating robotic vision-based navigation algorithms and are limited in scalability and their ability to provide perceptually real-time simulation. In addition, other frameworks are mostly based on classic simulation environments such as Gazebo[2], which are not necessarily as efficient, realistic, or scalable as modern engines such as Unity ([5], [3]). This is the reason for simulators becoming increasingly based on frameworks like ROS[4] and Unity([6], [7]). Hence, we chose to develop our own simulation framework that leverages the Unity game engine as a virtual environment simulator to take advantage of its advanced rendering pipeline, support for reliable physics engines and scalable memory management capabilities. ROS is used as the framework to model the autonomous/semi-autonomous robotic systems interacting with the virtual world in Unity.

II. SIMULATOR MODEL

The simulation environment uses Unity for simulating the environment, topography, photometry, rover dynamics and terrain interaction. The details of the modelling are in **(Ref.)**. The following sections give a brief overview of the same and system architecture.

*This work is directly supported by the NASA Solar System Exploration Research Virtual Institute cooperative agreement 80ARC017M0006.

[1]Center for Astrophysics and Space Astronomy, University of Colorado Boulder

[2]Department of Computer Science, University of Colorado, Boulder

[3]ATLAS Institute, University of Colorado, Boulder

A. Environment model

The lunar environment is modeled as multiple interacting sub-systems inside Unity environment. using high-resolution synthetic terrain

1) Topographic Model:

2) Photometric Model: Include Skymap

3) Rover Model:

4) Terrain Interaction Model:

B. Agent Model

In ROS.

C. Timer Model

1) Time Synchronization:

2) Real Time compliance:

3) Fixed Update:

III. Test System model

A. Test-Case generation

IV. Simulation

V. Results and Discussion

VI. Conclusion

Before you begin to format your paper, first write and save the content as a separate text file. Keep your text and graphic files separate until after the text has been formatted and styled. Do not use hard tabs, and limit use of hard returns to only one return at the end of a paragraph. Do not add any kind of pagination anywhere in the paper. Do not number text heads-the template will do that for you.

Finally, complete content and organizational editing before formatting. Please take note of the following items when proofreading spelling and grammar:

A. Abbreviations and Acronyms

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

B. Units

- Use either SI (MKS) or CGS as primary units. (SI units are encouraged.) English units may be used as secondary units (in parentheses). An exception would be the use of English units as identifiers in trade, such as 3.5-inch disk drive.
- Avoid combining SI and CGS units, such as current in amperes and magnetic field in oersteds. This often leads to confusion because equations do not balance dimensionally. If you must use mixed units, clearly state the units for each quantity that you use in an equation.
- Do not mix complete spellings and abbreviations of units: Wb/m2 or webers per square meter, not webers/m2. Spell out units when they appear in text: . . . a few henries, not . . . a few H.
- Use a zero before decimal points: 0.25, not .25. Use cm3, not cc. (bullet list)

C. Equations

The equations are an exception to the prescribed specifications of this template. You will need to determine whether or not your equation should be typed using either the Times New Roman or the Symbol font (please no other font). To create multileveled equations, it may be necessary to treat the equation as a graphic and insert it into the text after your paper is styled. Number equations consecutively. Equation numbers, within parentheses, are to position flush right, as in (1), using a right tab stop. To make your equations more compact, you may use the solidus (/), the exp function, or appropriate exponents. Italicize Roman symbols for quantities and variables, but not Greek symbols. Use a long dash rather than a hyphen for a minus sign. Punctuate equations with commas or periods when they are part of a sentence, as in

$$\alpha + \beta = \chi \tag{1}$$

Note that the equation is centered using a center tab stop. Be sure that the symbols in your equation have been defined before or immediately following the equation. Use (1), not Eq. (1) or equation (1), except at the beginning of a sentence: Equation (1) is . . .

D. Some Common Mistakes

- The word data is plural, not singular.
- The subscript for the permeability of vacuum ?0, and other common scientific constants, is zero with subscript formatting, not a lowercase letter o.
- In American English, commas, semi-/colons, periods, question and exclamation marks are located within quotation marks only when a complete thought or name is cited, such as a title or full quotation. When quotation marks are used, instead of a bold or italic typeface, to highlight a word or phrase, punctuation should appear outside of the quotation marks. A parenthetical phrase or statement at the end of a sentence is punctuated outside of the closing parenthesis (like this). (A parenthetical sentence is punctuated within the parentheses.)
- A graph within a graph is an inset, not an insert. The word alternatively is preferred to the word alternately (unless you really mean something that alternates).
- Do not use the word essentially to mean approximately or effectively.
- In your paper title, if the words that uses can accurately replace the word using, capitalize the u; if not, keep using lower-cased.
- Be aware of the different meanings of the homophones affect and effect, complement and compliment, discreet and discrete, principal and principle.
- Do not confuse imply and infer.
- The prefix non is not a word; it should be joined to the word it modifies, usually without a hyphen.
- There is no period after the et in the Latin abbreviation et al..

- The abbreviation i.e. means that is, and the abbreviation e.g. means for example.

VII. USING THE TEMPLATE

Use this sample document as your LaTeX source file to create your document. Save this file as **root.tex**. You have to make sure to use the cls file that came with this distribution. If you use a different style file, you cannot expect to get required margins. Note also that when you are creating your out PDF file, the source file is only part of the equation. *Your TeX $\rightarrow$ PDF filter determines the output file size. Even if you make all the specifications to output a letter file in the source - if your filter is set to produce A4, you will only get A4 output.*

It is impossible to account for all possible situation, one would encounter using TeX. If you are using multiple TeX files you must make sure that the “MAIN“ source file is called root.tex - this is particularly important if your conference is using PaperPlaza’s built in TeX to PDF conversion tool.

A. Headings, etc

Text heads organize the topics on a relational, hierarchical basis. For example, the paper title is the primary text head because all subsequent material relates and elaborates on this one topic. If there are two or more sub-topics, the next level head (uppercase Roman numerals) should be used and, conversely, if there are not at least two sub-topics, then no subheads should be introduced. Styles named Heading 1, Heading 2, Heading 3, and Heading 4 are prescribed.

B. Figures and Tables

Positioning Figures and Tables: Place figures and tables at the top and bottom of columns. Avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be below the figures; table heads should appear above the tables. Insert figures and tables after they are cited in the text. Use the abbreviation Fig. 1, even at the beginning of a sentence.

TABLE I

AN EXAMPLE OF A TABLE

One	Two
Three	Four

We suggest that you use a text box to insert a graphic (which is ideally a 300 dpi TIFF or EPS file, with all fonts embedded) because, in an document, this method is somewhat more stable than directly inserting a picture.

Fig. 1. Inductance of oscillation winding on amorphous magnetic core versus DC bias magnetic field

Figure Labels: Use 8 point Times New Roman for Figure labels. Use words rather than symbols or abbreviations when writing Figure axis labels to avoid confusing the reader. As an example, write the quantity Magnetization, or Magnetization, M, not just M. If including units in the label, present them within parentheses. Do not label axes only with units. In the example, write Magnetization (A/m) or Magnetization A[m(1)], not just A/m. Do not label axes with a ratio of quantities and units. For example, write Temperature (K), not Temperature/K.

VIII. CONCLUSIONS

A conclusion section is not required. Although a conclusion may review the main points of the paper, do not replicate the abstract as the conclusion. A conclusion might elaborate on the importance of the work or suggest applications and extensions.

APPENDIX

Appendixes should appear before the acknowledgment.

ACKNOWLEDGMENT

The preferred spelling of the word acknowledgment in America is without an e after the g. Avoid the stilted expression, One of us (R. B. G.) thanks . . . Instead, try R. B. G. thanks. Put sponsor acknowledgments in the unnumbered footnote on the first page.

References are important to the reader; therefore, each citation must be complete and correct. If at all possible, references should be commonly available publications.

REFERENCES

[1] M. Allan, U. Wong, P. M. Furlong, A. Rogg, S. McMichael, T. Welsh, I. Chen, S. Peters, B. Gerkey, M. Quigley, *et al.*, “Planetary rover simulation for lunar exploration missions,” in *2019 IEEE Aerospace Conference*. IEEE, 2019, pp. 1–19.

[2] N. Koenig and A. Howard, “Design and use paradigms for gazebo, an open-source multi-robot simulator,” in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, vol. 3. IEEE, 2004, pp. 2149–2154.

[3] A. Konrad, “Simulation of mobile robots with unity and ros: A case-study and a comparison with gazebo,” 2019.

[4] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, “Ros: an open-source robot operating system,” in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.

[5] U. G. Engine, “Unity game engine-official site,” *Online][Cited: October 9, 2008.] http://unity3d. com*, pp. 1534–4320, 2008.

[6] E. Babaians, M. Tamiz, Y. Sarti, A. Mogoei, and E. Mehrabi, “Ros2unity3d; high-performance plugin to interface ros with unity3d engine,” in *2018 9th Conference on Artificial Intelligence and Robotics and 2nd Asia-Pacific International Symposium*. IEEE, 2018, pp. 59–64.

[7] M. Bischoff, “ROS#,” June 2019. [Online]. Available: https://github.com/siemens/ros-sharp/releases/tag/v1.5