Cheap Autonomous Rovers for Multi-Agent Applications

*Freedom Rover Units

1st Jordy A. Larrea Rodriguez
Department of Electrical and Computer Engineering
University of Utah
Salt Lake City, USA
Jordy.larrearodriguez@gmail.com

2nd Brittney L. Morales
Department of Electrical and Computer Engineering
University of Utah
Salt Lake City, USA
brittneymrls@gmail.com

3rd Misael Nava
Department of Electrical and Computer Engineering
University of Utah
Salt Lake City, USA
misaelnava812@gmail.com

***Abstract*—The state of the art in autonomous swarms employs a decentralized model consisting of multi-agent networks. These robotic collaborative systems hold the potential to adapt to new environments and optimize individual performance to specific tasks without having to deal with global systems prone to single points of failure. Our team's focus lies therein in developing a multi-agent system capable of a decentralized network. The swarm will incorporates three two-wheel differential drive rovers interfaced through the Robot Operating System (ROS) via wifi telecommunication through the micro-ROS agent service. The central bay system consists of a single laptop to communicate objectives for the agents to complete (carefully designed demos). Our development stack will leverage ROS for project management, simulation capabilities, navigation libraries, and native server-client model in robotics applications. The rovers AI will incorporate simultaneous localization and mapping (SLAM) techniques for RT positioning based on a priori grid or map; thus, facilitating navigation through improved state space mapping. Thus, we introduce FRU-bot, an autonomous platform capable of deployment in decentralized and centralized systems.**

***Index Terms*—Decentralized Communication, Swarm Communication, Multi-agent, ROS, Gazebo**

I. Introduction

[Brittney]

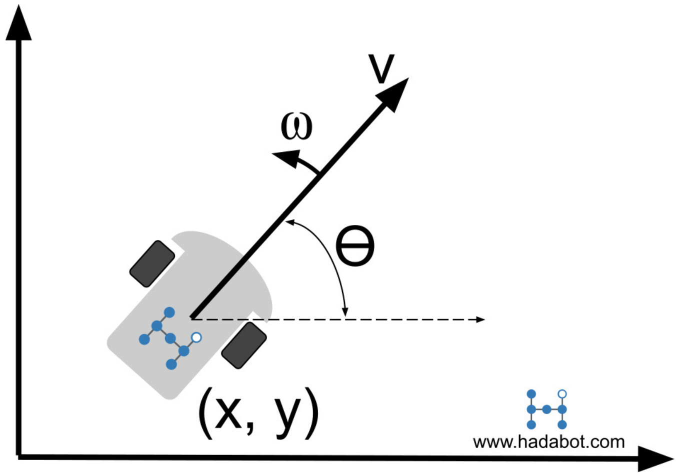

Fig. 1. Graphical representation of 2D robot pose information standard to 2 wheel robots [1].

II. Background

A plethora of elements must come together to successfully deploy a swarm of decentralized agents. Our system requires a working knowledge in both the physical and abstract. The agents are subject to their physical characteristics: i.e., their circuitry, power consumption, mechanical components, sensors, processing power, and geometry to navigate and respond to an environment. The complexity that drives cyber-physical systems is what largely makes robotics difficult. An agent must be able to function perfectly in a world with a continuous state space. Today, researchers are seeing a high degree of success with robots that incorporate mathematical and data driven modeling that leverages the superior computational capabilities of modern processors to solve complex continuous problems in robotics.

A. *Common Robot Autonomous Platforms*

Common mobile platforms for autonomous robots encompass a range of designs tailored to diverse applications within the field of robotics. Wheeled platforms, such as differential drive and omnidirectional robots, are popular choices due to their simplicity and efficiency in navigating various environments. Differential drive platforms use two powered wheels, allowing the robot to turn and move forward or backward by adjusting the speeds of the wheels. Omnidirectional platforms, on the other hand, utilize special wheel configurations to achieve unrestricted movement in any direction. Lastly, tracked platforms offer enhanced traction and stability, making them suitable for challenging terrains.

One notable platform is the TurtleBot 3, a popular open-source robot platform widely used for research and education. TurtleBot 3, developed by ROBOTIS, is a compact and modular robot equipped with various sensors, including a 360-degree lidar sensor, a camera, and inertial measurement units (IMUs). Its differential drive system allows for omnidirectional movement, making it versatile for navigation in dynamic environments. The TurtleBot 3 is often employed in the development and testing of autonomous navigation algorithms, mapping techniques, and obstacle avoidance strategies. Its affordability, ease of use, and vibrant community support make it an attractive choice for researchers, students, and hobbyists interested in exploring the intricacies of autonomous robotics.

B. *Robot Localization*

Normally, robots use a wide range of sensors to inform themselves about the surrounding environment. Our agents, for example, rely on a 6 axis inertial measurement unit (IMU) and rotatory encoders to produce an $\{X, Y, \theta, \dot{X}, \dot{Y}, \dot{\theta}, \ddot{X}, \ddot{Y}\}$ state configuration at every time step by considering physical properties and mechanics (Fig. 1). However, a frequent problem with using rudimentary techniques solely for odometry is that error propagated at each time step due to environmental conditions, sensor specifications, electrical noise, and rounding error can deviate a naive agent during localization. The problem persists even with the use of a known map and highly accurate sensors. Consider how a human might traverse

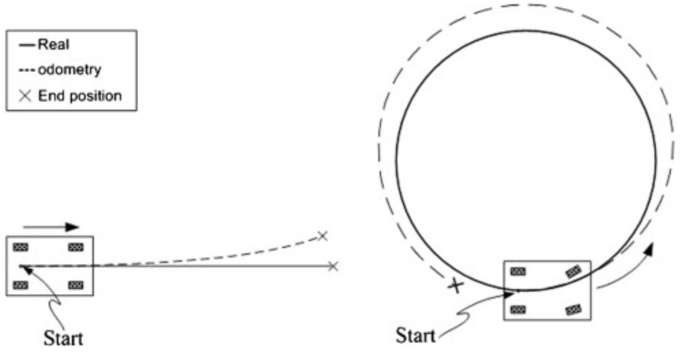

Fig. 2. Graphical representation of basic odometry error as a robot follows a reference trajectory [7].

a city by using a map. The human could ensure their own position by simply pondering their place in respect to features of the map like buildings, intersections, or street signs. Even if the human absent mindlessly navigated the city, they could triangulate their location by simply calculating their positions based on observations (such as the coffee shop across the street). Our robots, incorporate a Kalman ExFor an agent, SLAM leverages known knowledge of poses at past time steps and emissions or observations (sensor readings) made as the agent accumulates error through basic odometry to calculate probabilities of where the agent can be in space. Any number of sensors that detect 'emissions' from the environment can be used for SLAM; however, 2D lidar and cameras are regularly used for the high degree of precision awarded. Agents are capable of both learning a new environment and updating a known map with SLAM; thus, justifying our deployment of SLAM to improve the navigation capabilities of each agent.

C. Robot Kinematics

¡Jordy¿

D. ROS2 and Open Source Frameworks

¡Jordy¿

E. Robot Description and Transformation Trees

¡Jordy¿

F. Gazebo for Robot Simulation

¡Jordy¿

G. Autonomous Navigation

¡Jordy¿

H. Multi-Agent Swarm Systems

¡Jordy¿

III. Project Implementation

A. Prelminary Robot Chassis

B. Preliminary Electronics Design and Testing

C. Preliminary ROS2 Robot Description Design and Simulation

D. Hardware/Firmware

E. Multi-Agent Implementation with ROS2

IV. Evaluation

A. Conclusion

B. Appendices

V. Discoveries and Pitfalls

References

[1] Jack, "Implement ros2 odometry using vscode running in a web browser," 05/05/2023 2022. [Online]. Available: https://blog.hadabot.com/implement-ros2-odometry-using-vscode-in-web-browser.html

[2] W. Chen, J. Liu, H. Guo, and N. Kato, "Toward robust and intelligent drone swarm: Challenges and future directions," *IEEE Network*, vol. 34, no. 4, pp. 278–283, 2020.

[3] R. Cooley, S. Wolf, and M. Borowczak, "Secure and decentralized swarm behavior with autonomous agents for smart cities," in *2018 IEEE International Smart Cities Conference (ISC2)*. IEEE, Conference Proceedings, pp. 1–8.

[4] D. S. Drew, "Multi-agent systems for search and rescue applications," *Current Robotics Reports*, vol. 2, pp. 189–200, 2021.

[5] M. S. Güzel, V. B. Ajabshir, P. Nattharith, E. C. Gezer, and S. Can, "A novel framework for multi-agent systems using a decentralized strategy," *Robotica*, vol. 37, no. 4, pp. 691–707, 2019.

[6] S. Engebråten, K. Glette, and O. Yakimenko, "Field-testing of high-level decentralized controllers for a multi-function drone swarm," in *2018 IEEE 14th International Conference on Control and Automation (ICCA)*. IEEE, Conference Proceedings, pp. 379–386.

[7] K. Lee, W. Chung, and K. Yoo, "Kinematic parameter calibration of a car-like mobile robot to improve odometry accuracy," *Mechatronics*, vol. 20, no. 5, pp. 582–595, 2010.

[8] "Wiki," Aug 2018. [Online]. Available: http://wiki.ros.org/ROS/Introduction

[9] "Installation." [Online]. Available: https://docs.ros.org/en/humble/Installation.html

[10] "Understanding nodes." [Online]. Available: https://docs.ros.org/en/humble/Tutorials/Beginner-CLI-Tools/Understanding-ROS2-Nodes/Understanding-ROS2-Nodes.html

[11] "esp32-wroom-32e_esp32-wroom-32ue_datasheet_en_2023," 2023. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf

[12] F. Finocchiaro, "micro-ros porting to esp32," Aug 2020. [Online]. Available: https://micro.ros.org/blog/2020/08/27/esp32/

[13] P. Garrido, "micro-ros humble hawksbill release," May 2022. [Online]. Available: https://micro.ros.org/blog/2022/05/31/humble_release/

[14] R. C. Cardoso and A. Ferrando, "A review of agent-based programming for multi-agent systems," *Computers*, vol. 10, no. 2, p. 16, 2021.

[15] T.-K. Hu, F. Gama, T. Chen, Z. Wang, A. Ribeiro, and B. M. Sadler, "Vgai: End-to-end learning of vision-based decentralized controllers for robot swarms," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, Conference Proceedings, pp. 4900–4904.

[16] S. Khaliq, S. Ahsan, and M. D. Nisar, "Multi-platform hardware in the loop (hil) simulation for decentralized swarm communication using ros and gazebo," in *2021 IEEE 22nd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE, Conference Proceedings, pp. 310–315.

[17] P. McEnroe, S. Wang, and M. Liyanage, "A survey on the convergence of edge computing and ai for uavs: Opportunities and challenges," *IEEE Internet of Things Journal*, 2022.

[18] A. Pandit, A. Njattuvetty, and A. K. Mulla, “Ros-based multi-agent systems control simulation testbed (mascot),” *arXiv preprint arXiv:2212.12657*, 2022.

[19] S. Qamar, S. H. Khan, M. A. Arshad, M. Qamar, and A. Khan, “Autonomous drone swarm navigation and multi-target tracking in 3d environments with dynamic obstacles,” *arXiv preprint arXiv:2202.06253*, 2022.