ICE-RASSOR: Intelligent Capabilities Enhanced Regolith Advanced Surface Systems Operations Robot

Michael A. DuPuis[1], Joseph M. Cloud[1], Evan A. Bell[1], Andrew J. Nick[1], Kurt W. Leucht[1], Jonathan D. Smith[1], Jennifer G. Wilson[1], Brad C. Buckles[2], Thomas J. Muller[2]

1) NASA Kennedy Space Center, KSC, FL 32899, USA: {michael.a.dupuis, joe.cloud}@nasa.gov
2) LASSO/Kennedy Space Center, KSC, FL 32899, USA

Introduction

NASA's Regolith Advanced Surface Systems Operations Robot (RASSOR) is designed to mine and deliver regolith for In-Situ Resource Utilization (ISRU) processing. RASSOR's design enables it to collect and deposit regolith, return collected material for processing, and myriad related ISRU activities. To reliably perform these operations on the lunar surface, RASSOR software and sensory systems need to be robust and maximize the information extracted from on-board sensing. Currently, RASSOR lacks the ability to estimate collected regolith mass, and digging controllers are inefficient and unsuitable for general autonomous operation.

Figure: RASSOR 2.0 in the 'Big Bin', KSC's regolith test bed.

Drum Mass Estimation

We trained a neural network to model a relationship between actuator sensor values and known regolith masses. For data collection, RASSOR was secured to a stand and weights were progressively added to the bucket drum's geometric center while recording the actuator joints sensor data. We collected data in two scenarios. In one case (static), RASSOR's arm was held stationary at 0 degrees to horizontal, while in the other case (dynamic), the arm performed an up/down motion with different weights to capture relationships between joint angle, velocities, and the actuator current measurements. In all, data for 17 dynamic weight trials were recorded, while 38 static weight measurement data sets were recorded. The weights on the static data ranged from 0.0 lbs to 50.0 lbs. All robot sensor channels were recorded, though only kinematic and system power data was used.

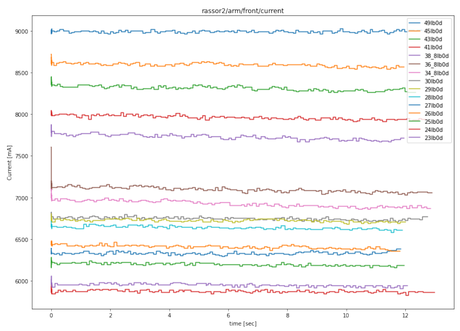

Figure: Raw static data collecting from RASSOR current sensors.

As training input, the model was provided the current draw in milliamps for the static data and the corresponding mass values as output. The mean squared errors were calculated between the ground truth data and the predicted values from regression. With the outliers removed, the error was less than 1.3 kg^2, with outliers, the error measurements peaked at 12.7 kg^2. Additional data is necessary to produce a meaningful result for the dynamic model.

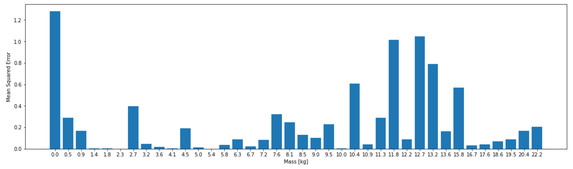

Figure: Mean-squared error of the learned model at different weights.

Excavation Simulation Development

The goal of reinforcement learning is for an agent to learn a policy (task strategy) through interactions with an environment. When the agent performs an action, a change occurs in environment state and a numerical reward is received which informs the agent whether the action performed was good or not. Since reinforcement learning algorithms learn through trial-and-error, a simulation is a desirable first environment for development and learning. Here, we show our 2D excavation simulation developed to facilitate learning.

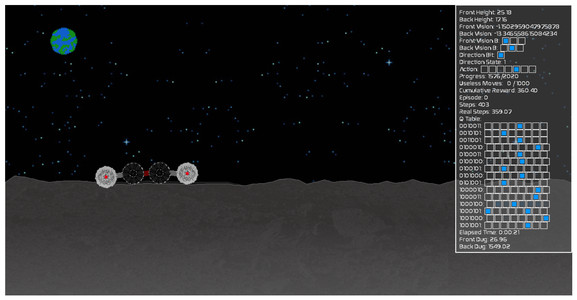

Figure: 2D simulation displaying RASSOR agent on surface and HUD.

This 2D simulation did not model physics in the environment. Instead, the surface is represented by a spline that is perturbed when intersected by RASSOR's drums. The robot wheels are affixed to the spline so "acrobatic" effects are not a current capability, though motion is visualized when the robot is moving (wheel and drum rotation). This simulation serves as a deterministic testing environment for the reinforcement learning algorithms, and as a precursor for the 3D environment and possible future physical deployment of the algorithms. On an i9-10900X CPU, the system can process 105 simulation steps/second in single-threaded mode, and over 2000 steps/second combined in multi-process concurrent mode.

Learning to Excavate

Q-learning was the reinforcement learning algorithm used to learn simplified excavation strategies in the 2D simulation. The action strategy for Q-learning is based on iterating over all possible action values within a given state and selecting the action that results in the largest action value (actions include: move left/right, move drums up/down to dig).

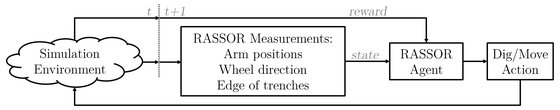

Figure: ICE-RASSOR reinforcement learning loop.

We experimented with various virtual sensor payloads to identify a combination that enabled efficient excavation operation and learning. We implemented pseudo time-of-flight sensors to report distance from each drum to ground and the height above ground which was found to be more efficient than other methods such as dividing the environment into a grid world to represent state.

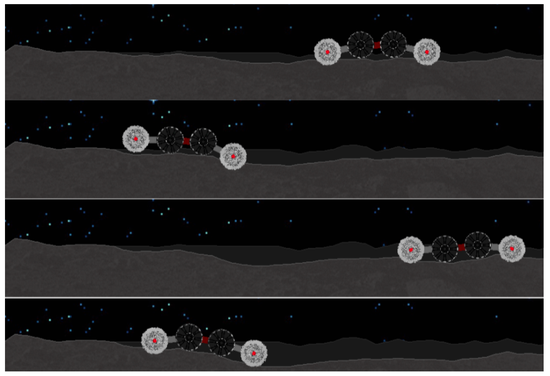

Figure: RASSOR shown at various steps of training.

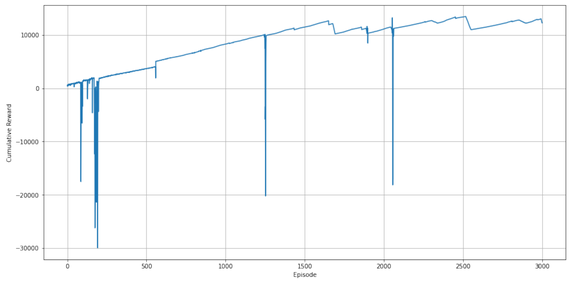

Figure: Reward score over 3000 episodes.

The figures above show RASSOR's terminal progress across learning episodes, and the cumulative reward that the agent received throughout training. The reward function is based on how much material is excavated per step. A penalty is also received for leaving the dig site or for inefficiencies such as rapid arm servoing.

Future Work

Future work includes regolith mass estimation during dynamic operation and expanding our simulation to more complex environments that facilitate transfer learning from simulation to a physical agent in KSC's regolith test bed.

Figure: 3D simulation developed using game physics engine.

Initial progress of our 3D simulation development is shown above. This simulation leverages Unity's physics engine to simulate simplified soil interactions and increased fidelity of the dynamic models of the robotic agents. Improved simulation fidelity enables a wide range of new capabilities, including the development and training of additional sensing capabilities, and research both at the granular mechanics and operations levels.

Discussion and Conclusion

A key research question in this work is: what is the minimal sensory payload that could enable RASSOR to carry out its operations autonomously? We applied supervised learning using real data to estimate the soil mass collected without the need for mass flow rate monitors or other explicate sensing techniques. We also create a reduced-order simulation environment to develop autonomous trenching controllers via reinforcement learning and prototype state estimation architectures. Our initial results suggest that excavated regolith mass can be inferred within 2.9% RMS error of full scale, and reinforcement learning for autonomous operations has learned trenching strategies and helped identify desirable sensing capabilities, positioning, and considerations.

Acknowledgements

This work is funded under a NASA Kennedy Space Center Independent Research and Technology Development (IR&TD) program.

References

[1] R.P. Mueller, J.D. Smith, J.M. Schuler, A.J. Nick, N.J. Gelino, K.W. Leucht, I.I. Townsend, and A.G. Dokos.
Design of an Excavation Robot: Regolith Advanced Surface Systems Operations Robot (RASSOR) 2.0.
In *2016 American Society of Civil Engineers (ASCE) Earth and Space Conference Proceedings*. ASCE, 2016.