Control Policy Design for Path Planning and Obstacle Avoidance Using Adpative Model Predictive Control of a Four-Wheel Based Extraterrestrial Rover

Jason Zhou{zzhou292@wisc.edu}, Yulong Yue{yyue32@wisc.edu}

May 7, 2023

1 Research Problem Statement

In this project, we proposed to design a rover path planning and following algorithm to provide an extraterrestrial rover navigation strategy. We propose to solve the problem by solving a closed-loop motion planning and control problem based on optimization with constraints. The control strategy will be benched in a simulated environment in Project Chrono. For the project, we will assume privileged information, such as locations, sizes, and shapes of the obstacles. Also, we will assume the environment is limited and fixed.

2 Background

2.1 Rover Control Strategy

Providing control solutions to extraterrestrial rover can always be challenging, due to the low tolerance to mistake and long signal transmission delay. Nowadays, most space exploration agencies, such as The National Aeronautics and Space Administration (NASA), are adopting two different ways for rover control:

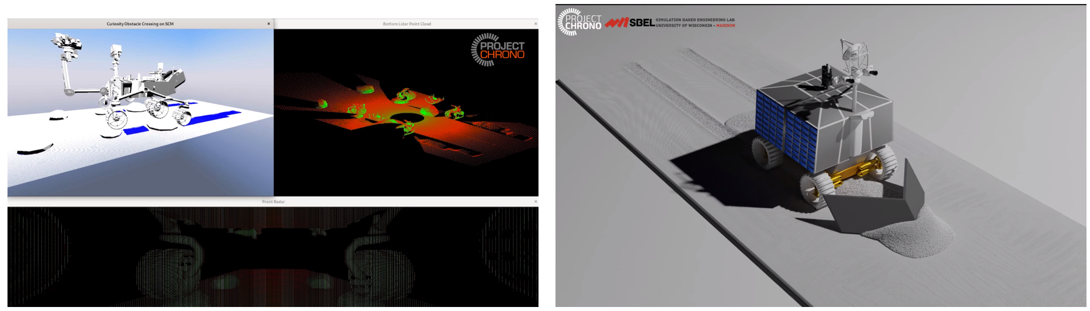
- Manual Explicit Motion Commands
- A propitiatory Autonomous Path Planning and Path Following Control Policies

The first manual control strategy is relatively simple. Based on the images transmitted from the rover, the ground base provides direct motion commands, such as ”move forward for 5m”, to the rover. The method relies heavily on human judgment, along with data and command transmission, which usually suffer from the signal delay between the ground base on earth and extra extraterrestrial planets. This control method will not be discussed in the project.

The second control method, although proprietary and few details have been disclosed by NASA, is an autonomous path planning and trajectory following control algorithm which matches the materials covered in the class. Based on the introduction provided by NASA, the ground base on earth, based on the images and data transmitted from the rover, sends a control signal containing the location of the next ’waypoint’. Then the rover, based on its own sensing capabilities using sensors such as camera and lidar, plans the route and tracking the planned trajectory. [1]

2.2 Simulation Through Project Chrono

Project Chrono is a multi-physics modeling and simulation infrastructure based on a platform-independent, open-source design. Project Chrono has been used widely for the simulation of autonomous vehicles and robotic systems. Project Chrono, in the past two years, received funding from NASA to provide simulation support to investigate wheel-terrain interaction of the 2024 NASA VIPER Moon Rover using different types of terra-mechanics simulation method, as shown in Figure 1(b) and Figure 1(a). Therefore, the software package already contains multiple ready-to-deploy extraterrestrial rover models and their control interfaces.



(a) Curiosiy Mars rover on SCM deformable terrain with lidar and radar sensing simulation

(b) VIPER Lunar rover conducting bulldozing experiment on granular terrain simulated by Smoothed-Particle Hydrodynamics (SPH) Method

Figure 1: Project Chrono contains complete rover models, control interfaces, and deformable terrain models for wheel-terrain interaction

In this project, we will develop the algorithm in Matlab, and use Project Chrono as our simulation environment. In the simulation, we will use the Soil-Contact Model (SCM) deformable terrain model to simulation soil deformation. The SCM model is a simple semi-empirical model which is built upon the original Bekker-Wong Equation, as shown in Equation 1. A more detailed description can be found in [2]. In the Equation 1, K_c, K_ϕ, and n are semi-empirical parameters defining the soil properties, z is the sinkage, b is the average wheel contact width, and p is the normal pressure applied to the rover wheel (to provide normal force in the simulation).

$$p = \left(\frac{K_c}{b} + K_\phi\right) z^n \,. \tag{1}$$

3 Proposed Solution

The problem is simplified into an optimal path planning and control optimization problem by assuming a square terrain with a center at (0.0, 0.0), and a fixed side length of 20 meters. Three different types of rocks with similar radius, while viewing birdeye view, are used. To further simplify the problem, a bounding circle, or what can be called a safety circle, is drawn around the rock. From a bird's eye view, looking from above, the problem environment is simplified into a 2D path planning and optimal control problem with a given square terrain, and with several circular-shaped obstacles, as shown in Figure 3.

We employ a dynamic model of a simplified rover as shown in Equations 2.

$$\begin{cases} \dot{x} = v(t)cos(\theta(t)) \\ \dot{y} = v(t)sin(\theta(t)) \\ \dot{v}(t) = a(t) \\ \dot{a}(t) = u_1(t) \\ \dot{\theta}(t) = \frac{v(t)tan(\phi(t))}{l_a} \\ \dot{\phi} = u_2(t) \end{cases} . \tag{2}$$

After the simplification, we assume that each obstacle has the properties shown in Equation 3, in which x_{obs_k} and y_{obs_k} represent the location of the k'th obstacle's bounding circle, and r_{obs_k} represent the radius of the bounding circle of the k'th obstacle. If we assume the entire time domain of the problem is t_f, and there are M obstacles presented in the environment. Assuming the starting location of the rover is [x_{start}, y_{start}], the given target location is [x_{end}, y_{end}], the initial rover yaw angle is yaw_{t_0}, and the initial rover steering angle 0. We formulate the initial condition, terminal condition, and path constraint for the optimal control problem:

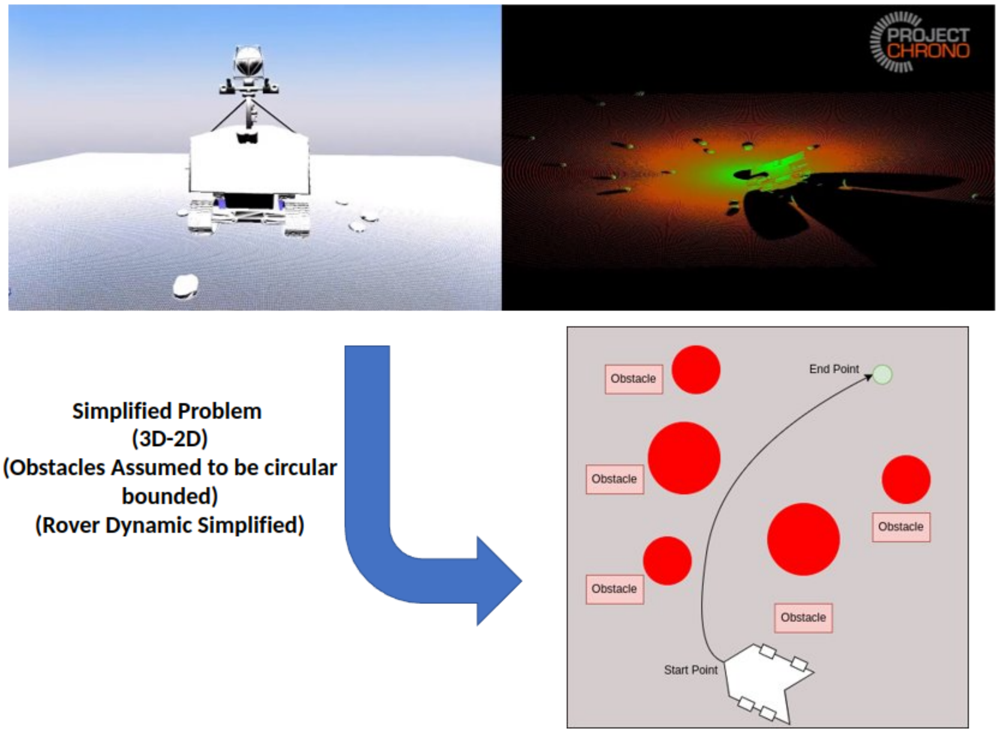


Figure 2: The problem will be simplified into a 2-dimensional path planning and optimal control problem. The optimization will be solved in the 2-D space and applied and benched in the 3-D simulation.

$$\begin{cases} x_0 = x_{start}, y_0 = y_{start} \\ x_{t_f} = x_{end}, y_{t_f} = y_{end} \\ \theta_0 = yaw_{t_0} \\ \phi_0 = 0 \\ r_i < (x_i - x_{obs_k})^2 + (y_i - y_{obs_k})^2 \\ 0 \leq i \leq t_f, 0 \leq k \leq M \end{cases} . \tag{3}$$

In addition to the above path constraints, an extra constraint is added to ensure on-board equipment's stability:

$$-0.4 \leq \frac{u_2(t)}{l_{axes} cos^2(\theta(t))} \leq 0.4 \tag{4}$$

In the above equation, the l_{axes} represents the distance between two axes of the rover. For the Viper lunar rover which is modeled in this project, the $l_{axes} = 1.4m$

The objective is for rover to finish the path in minimum time, as a results, the optimal control problem is formulated as:

$$\min_{x,u,t_f} t_f \tag{5}$$

After the optimization problem has been solved, a control input series u_{opt} and a control path series defined by x_{opt} and $yopt$ are expected to be the output. However, the control input series u_{opt} will need to be post-processed in order to feed into the Chrono simulation. This is due to the fact that the simplified rover dynamic equations defined in Equation 2 is a significantly simplified control, by inputting steering angle and rover velocity. However, the actual rover control is defined by wheel velocity, and the steering capability is limited by the steering motor's rotational speed. For example, it is possible that the velocity input from u_{opt} jumped instantaneously from 1 to -1, while in reality, this is unlikely to happen. We expect to use two PID controllers, as defined in Equation 6, to provide path following (steering) and vehicle velocity control. As a result, a path will be created in the simulation based on x_{opt} and y_{opt}, and a steering PID controller will be created for the rover to follow the path, and a speed PID controller will be used to control the rover speed.

$$u(t) = K_p e(t) + Ki \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \tag{6}$$

4 Limitation and Future Work

We assume the control algorithm can obtain privileged information, such as the size of the terrain, the exact locations and the sizes of the obstacles, and the rover location due to the time limitation of the course project. However, we plan to extend the research focus to replicate what happens in real-life engineering - the environment, obstacles information, and rover information are all acquired through sensing, such as camera sensors, lidar sensors, radar sensors, and potentially position sensors. As Project Chrono can simulate sensor data, such as shown in Figure 1(a), in the near future, we will conduct research to reconstruct the 3d environment and obtain the terrain and obstacle data from the ”real” environment.

5 Proposed Milestone Deadline

Date	**Item**
03/31/2023	Algorithm Implementation in Matlab
04/10/2023	Building the network communication between Matlab and Project Chrono
04/21/2023	Aetting up the simulation environment in Project Chrono
04/28/2023	Building of the control interface to provide waypoint commands to the controller
05/05/2023	Final testing with Matlab control and Project Chrono simulation
05/12/2023	Final deliverable and report

References

[1] K. Rainey, “Station astronauts remotely control planetary rover from space,” 2013.

[2] R. Serban, J. Taves, and J. Zhou, “Real-time Simulation of Ground Vehicles On Deformable Terrain,” *Journal of Computational and Nonlinear Dynamics*, pp. 1–10, 02 2023.