Design Methods for Cost-Effective Teams of Mobile Robots in Uncertain Terrain

by

Nathaniel Steven Michaluk

Bachelor of Science, Mechanical Engineering
Pennsylvania State University, 2012

Bachelor of Science, Nuclear Engineering
Pennsylvania State University, 2012

Submitted to the Department of Mechanical Engineering
in Partial Fulfillment of the Requirements for the Degree of

Master of Science in Mechanical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2014

Signature of Author: ....
Department of Mechanical Engineering
May 9, 2014

Certified by: ....
Richard M. Wiesman
Professor of the Practice of Mechanical Engineering
Thesis Supervisor

Accepted by: ....
David E. Hardt
Ralph E. and Eloise F. Cross Professor of Mechanical Engineering
Chairman, Committee on Graduate Theses

Design Methods for Cost-Effective Teams of Mobile Robots in Uncertain Terrain

by

Nathaniel Steven Michaluk

Submitted to the Department of Mechanical Engineering
on May 9, 2014 in Partial Fulfillment of the
Requirements for the Degree of Master of Science in
Mechanical Engineering

ABSTRACT

Conducting planetary exploration missions with mobile robots is expensive, with costs ranging from hundreds of millions to billions of dollars. Developing reliable robots to work remotely on rough, uncertain terrain is imperative for these missions. One potential tactic for improving the cost-effectiveness of these missions is to distribute the mass allowance for the mission over a team of smaller robots, rather than using a single robot. However, there is limited work on determining the size and design for a team of robots to provide the best overall performance when operating on hazardous terrain.

This thesis develops a framework for designing mass-restricted, homogenous teams of mobile robots that will operate in a region with uncertain terrain conditions. The framework is built around three models: a four-wheeled robot model, a probabilistic model of terrain hazards, and a robot-terrain interaction model. The models are formulated into an optimization problem that can be used to determine the best design for a team of robots based on the team's combined equivalent straight-line velocity (CESLV), a novel measure of mission performance. CESLV is an effective measure of mission performance for both predetermined (static) mission plans and dynamic mission plans, where observations made by the robots can change the future mission tasks. A graphical user interface (GUI) is also presented which allows a designer to explore the design tradespace for the team of robots while considering important factors that are not captured by the models.

In a case study of a Mars exploration mission, a team of robots provides superior performance to a single robot. A sensitivity analysis shows that the optimal size of the robot team is robust to inaccuracy in the terrain conditions. Additionally, the tradespace UI captures a trend in robot team design that would have otherwise gone unnoticed.

Thesis Supervisor: Richard M. Wiesman
Title: Professor of the Practice of Mechanical Engineering

Acknowledgments

I owe the most gratitude to my advisor, Dr. Richard Wiesman, for his help getting me started on this work. He has been insightful, understanding, and patient, and it has been a pleasure to work with him.

I thank the MIT Skoltech Initiative for supporting this research as part of the Space Exploration Strategic Development Project.

I want to thank my friends in the Field and Space Robotics Lab at MIT for their advice on my work, but perhaps more importantly for the company and comradery we have shared on a daily basis. Dr. Steven Dubowsky graciously helped me focus my work and further my professional development on many occasions.

I thank my parents and family for their support and encouragement during my entire education. My future wife, Alexandra Shoffner, has been supportive and caring. She has kept all of the other aspects of our lives in order on nights and weekends when I have been working.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Recent trends indicate that both the rate and complexity of missions to other terrestrial bodies, such as the moon or Mars, will increase in the next decade. NASA is currently involved in seven active Mars missions and is planning at least two additional missions that will launch by 2020. Across major space agencies there are approximately six missions planned within the next decade that incorporate a lander and a mobile robot ("rover") [4]. There are also more daring missions, such as Mars One's missions, which aim to establish a long-term human settlement on Mars.

The costs of robotic planetary exploration missions have increased by several orders of magnitude in the past two decades. Figure 1.1 shows a cost comparison between the three NASA missions that landed rovers on Mars. NASA is planning a Mars mission with a Curiosity-like rover in 2020 that is targeted to cost $1.5 billion [5]. The 2016 and 2018 ExoMars missions, which are headed by the European Space Agency (ESA) and include two landers and a rover, have an estimated cost of $1.6 billion [6].

One avenue for reducing the cost without sacrificing the productivity of future missions is to utilize teams of robots. Teams of robots retain the overall capabilities of the single robot that would otherwise be used. The benefit is that the individual robots within the team could either be much simpler than the alternative single robot or be more cost-efficiently sized for the terrain conditions.

In addition to cost, shifting from a single robot to a team of robots offers other potential advantages. Robot teams can provide improved reliability through redundancy and can provide flexible mission execution. The team can be transported via multiple smaller, staggered launches. A fraction of the team can perform work prior to other arrivals, or in the case of any failures. Additionally, robots could coordinate to perform tasks that are impractical for a single robot, such as surveying a construction site.

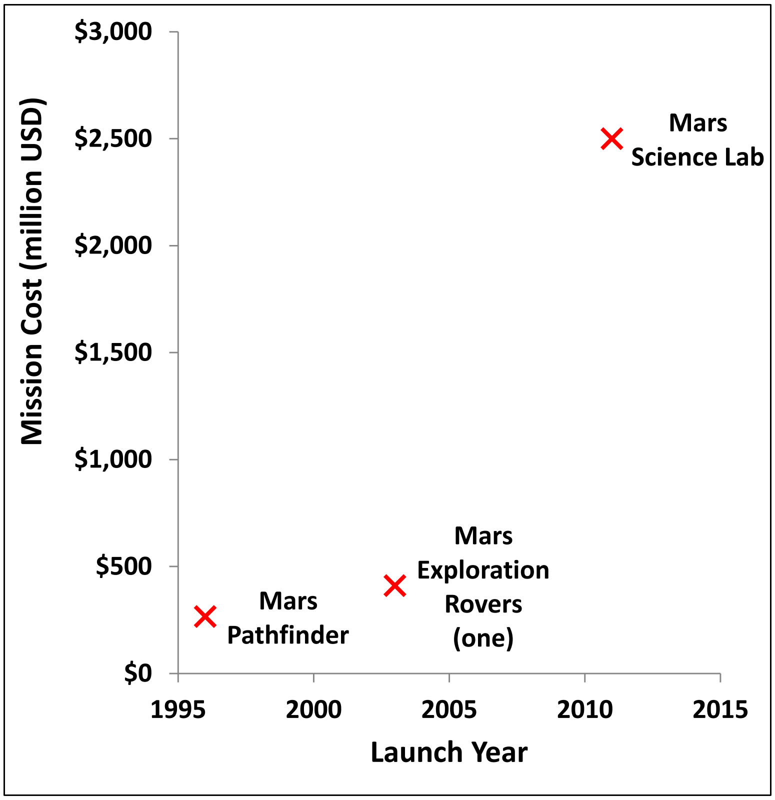


Figure 1.1: Cost history of NASA rover missions to Mars [1, 2, 3]

1.2 Problem Statement

This thesis examines the tradeoff between robot size and mobility performance to determine the optimal degree of fractionation for a mass-limited, homogeneous team of mobile robots operating in uncertain terrain. The design framework involves developing a mobile robot model, including models for the interaction of the robot with different terrain hazards. Statistical models represent the uncertainty of the terrain in the region where the robot would be operating. A novel metric of mobility and mission performance, referred to as the combined equivalent straight-line velocity (CESLV), helps describe how fast a given team could visit a given set of destinations. With these models, an optimization routine can be used to evaluate the most cost-effective design for a robot or a team of robots, and how a designer or a mission planner could adopt these methods to evaluate the tradespace of robot team designs for a given mission.

1.3 Scope

This thesis contributes a design framework to help determine how to allocate a given allowance of mass among a team of robots to achieve optimal or near-optimal mobility performance for a mission. Designing mobile robots for planetary exploration (as well as for other purposes) is a difficult problem due to complex, interdependent systems and constraints. Mobility of planetary robots is an especially challenging aspect of design [7]. The overall performance of a mobile robot depends on its locomotion system, its sensing and navigation systems, its communication system, and its control system. It is inevitable that some components of the robot will be over-designed with respect to others, and thus effort (in the form of mass or cost) is wasted.

Furthermore, the success of a planetary exploration mission also depends upon the effectiveness of the mission plan. Some missions will have a static mission plan, in which the tasks are set to be performed in a known order. In other cases, subsequent portions of the mission plan may change depending upon results from earlier portions of the mission. For these dynamic mission plans, the remaining tasks in the mission could change due to a key observation or discovery.

In order to focus on the design tradeoffs between robot mass and mobility, the models in this thesis are only as complex as they need to be in order to fill their part within the framework. Many of the models could be made more accurate based on the current state of the art, but at the cost of increased complexity. Additionally, a designer could supplement the models to account for other factors or tradeoffs in mobile robot design. The designer needs to determine what level of complexity is appropriate for a given situation. This thesis makes several assumptions to limit the complexity of the problem.

First, the mission plan can be split into two parts: travel and functional tasks. Travel consists of the robot driving from one location of interest to another. Functional tasks consist of the operations that the robot performs at locations of interest that directly contribute to the mission goals. For example, the tasks could include capturing pictures, taking measurements with scientific instruments, or collecting soil samples for analysis. This framework focuses on the ability of the robot to travel between locations of interest, and assumes that the ability of the robot to perform any functional tasks is separate and unrelated to mobility. The only exception to this assumption is that the mass of any equipment for performing function tasks acts as a payload on the robot.

Second, this framework makes reasonable assumptions regarding the capabilities of the robot's control software and navigation system. It assumes that the robot's navigation capabilities are similar to that of NASA's Mars rovers. Specifically, the robot is able to detect if terrain obstacles are safe to traverse, and thus avoid immobilization, failure, or damage due to terrain conditions. Additionally, the travel speed of the robot is limited by the speed at which it can process upcoming terrain conditions.

Third, this framework assumes that there are no component failures in the robots. This assumption is based on the reliability of NASA's Mars rovers.

Fourth, the cost of the team of robots is measured by their total mass. Attempting to develop cost models for planetary robot components would be difficult, and any models would likely be inaccurate, due to the specialized hardware that is required for extraplanetary operations. Instead, this framework uses the mass of robots to represent their cost. Especially for planetary exploration, mass is an effective indicator of mission cost. Transportation costs, including the costs of the launch, the spacecraft, and the lander, tend to dominate the overall mission cost. Additionally, this framework can build off of existing work on mass modeling of different planetary rover components

[8, 9, 10, 11, 12].

1.4 Background and Literature Review

During the past three decades, research on mobile robots has been popular across a variety of fields. Most of this work is relevant to planetary exploration robots, with a significant portion focusing directly on planetary rovers. Areas of research include hardware technology, control, locomotion systems, terramechanics, navigation, mission planning, and mission simulation, among others. This section presents previous work that has had a profound impact on the work in this thesis, including trending research on teams of mobile robots (Section 1.4.1), the interaction between wheels and terrain (Section 1.4.2), and mobile robot design (Section 1.4.3).

1.4.1 Teams of Mobile Robots

Using a team of mobile robots for extraplanetary missions has been considered since the 1970s [13]. Researchers have considered the merits of small teams, consisting of fewer than five robots, and large teams, consisting of several dozen robots. Research on small teams has examined multi-robot cooperation on single tasks, such as assembling structures or for improved mobility on hazardous terrain [14, 15, 16]. Research on large teams, or "swarms," has examined navigation, area coverage, formations, and target tracking [17, 18, 19, 20].

The concept of robot teams captures much attention from the research community not only for the unique capabilities of teams, but also for their potential cost-savings and potentially superior performance. The use of cooperative multi-robot systems represents a paradigm shift from the use of single agents, for simplicity, to teams of multiple agents, which in many cases may benefit from the inherent structure of the mission scenario. Despite individual agents possibly becoming simpler, robotic systems will maintain and improve upon their net capabilities through improvements to autonomy and cooperation. The simplification of individual robots offers potential for lower-cost, rapid production of robots using off-the-shelf components [21].

In addition to their potential cost-savings, teams are likely to be more reliable and more robust than a single robot with comparable overall capabilities. Robot teams could still perform all their functions, possibly at a decreased rate, even with the failure of a portion of their agents. Teams may also be better able to cope with partial failures of individual agents. For example, the partially-functional robots could be assigned to a task that does not require full functionality. In an ideal scenario, damaged or malfunctioning robots could be repaired by their team members.

Stancliff et al. performed the most notable work on evaluating the benefits of these concepts and determining how to best use them [22]. They examine methods to optimally design teams of planetary exploration robots based on the reliability of the robots' components. They consider a cost model based on the reliability of robot components, and compare the tradeoff between team size and individual reliability of each robot. Stancliff et al. also include the possibility that robots can be repaired by their teammates after they break. Their design methods rely on Monte-Carlo simulations to evaluate mission performance of different robot team designs. A few others have done similar work examining the tradeoffs between team size and individual reliability [23, 24, 25].

1.4.2 Terramechanics for Mobile Robots

The design of mobile robots depends heavily on the interactions between the terrain and the running gear of the robot. Bekker presents what is now the fundamental theory of vehicle-soil interaction, often referred to as "Bekker theory" [26, 27]. Parts of this theory build off of Terzaghi's work on soil mechanics [28, 29]. Bekker includes vehicle-soil interaction analyses for wheels and tracks, as well as discussions on other types of locomotion. Wong has since then updated and supplemented Bekker's work [30]. The work of Bekker and Wong is essential for determining the tractive and resistive forces generated by mobile robots on off-road terrain.

Despite this theory being developed for the larger part of the past century, it remains the subject of experimental research. Ding et al. and Sutoh et al. have recently performed experimental work to further investigate the effect of wheel design, including grouser design, on traction and resistance in loose soils [31, 32]. Skonieczny et al. and Liu et al. investigate the effect of grousers in soil, which to this point has received limited theoretical research [33, 34].

In addition to the vehicle-soil mechanics, the effects of various terrain obstacles must be considered. Any terrain feature that may impede the forward movement of a vehicle, beyond the resistance generated by the running gear moving through the soil, can be considered an obstacle. Four generalized obstacles are typically considered in mobility analyses: a slope, a step, a bump (also referred to as a "wall"), and a ditch (or "trench"). Apostolopoulos's work analyzes slope, step, and bump climbing for a wheeled vehicle [35]. More recently, Berkemeier et al. analyze step climbing for a wheeled vehicle [36]. Rajabi and BSC analyze step climbing for a tracked vehicle [37, 38].

1.4.3 Mobile Robot Design Methods

To a large degree, the models in this thesis are successful due to the quality of earlier research. Several efforts have been made to design the locomotion configuration of mobile robots for a specific or optimal mobility. These studies, listed and described below, employ Bekker theory along with other methods to determine mobility characteristics. The designs are evaluated with respect to a number of mobility metrics.

Apostolopoulos's work represents one of the initial efforts to design a mobile robot based on an analysis of terramechanics and significantly influences this research [35]. He considers three mobility performance measures: trafficability, maneuverability, and terrainability. Trafficability is the ability to traverse ground (soil, pavement, gravel, etc.) without losing traction. Maneuverability is the ability to navigate around obstacles in an environment. Terrainability is the ability to cross terrain obstacles. To investigate trafficability and terrainability, Apostolopoulos analytically determines drawbar pull from wheel-soil interactions, while considering resistances produced by obstacles including slopes, steps, and bumps. Maneuverability is determined from the skid steering resistance of the robot. He evaluates the design of robots based on metrics from these three categories.

Zhang et al. develop a method to evaluate the mobility performance of a mobile robot with the Rocker-Bogie suspension [39]. They also determine drawbar pull, stability on slopes, and obstacle crossing ability, but they uniquely include the effect of grousers in the sinkage of the wheels. The robot's obstacle-crossing ability is determined through empirical relations developed from experiments and simulations.

Patel, Michaud, and Thueer develop methods to evaluate and design robots based on certain mobility metrics. Patel et al. developed RMPET (the rover mobility performance evaluation tool), which analytically determines traction and resistances to calculate drawbar pull [40]. The tool also calculates the mean free path for the robot based on terrain statistics. Wilcox et al. first propose using the mean free path, the average distance the robot can travel without encountering an untraversable obstacle, as a measure of how well a robot would perform on specific terrain [41].

Michaud et al. develop the rover chassis evaluation tools (RCET), which encompasses the methods of Patel et al. used in RMPET [42]. This tool also determines robot stability, gradeability, and obstacle-climbing ability. Note that obstacle-climbing ability is compared in terms of the friction coefficient needed to climb the obstacle, rather than by specifying the maximum obstacle that can be traversed.

Thueer and Seigwart evaluate the mobility of robots using supplemental parameters to those used by Apostolopoulos [43]. For a robot, they determine the minimum wheel-soil friction coefficient to avoid slipping, the peak torque required by the motors to traverse an obstacle, and the accumulated slip over a simulated driving distance. They also propose a new metric called the "velocity constraint violation," which represents the risk of violating the kinematic constraints of the robot based on ideal wheel velocities, which would result in slippage.

Lamamy develops a robot model that attempts to relate the cost of a robot to its performance in an exploration mission [8]. This is intended to allow for the generation of a tradespace and the selection of a robot that provides certain mission results. The robot modeling includes major subsystems of the robot (such as thermal, power, and communications subsystems), and has been especially useful in developing this thesis. The mobility portion of the model determines the maximum speed of the robot on flat ground and the maximum obstacle (rock) size that can be climbed. However, the mobility analysis is limited, relying heavily on empirical relationships.

Alibay extends Lamamy's work to design teams of planetary robots [11]. Her modeling incorporates some of Lamamy's work and adds a terramechanical analysis to determine parameters such as drawbar pull and sinkage. Additionally, she analyzes missions based on their required functional tasks. She then determines the possible ways that robots in a team could be equipped with the different hardware required for the tasks. The teams are evaluated in terms of metrics that aim to capture 1) their effectiveness to complete tasks based on the functional breakdown of the team, 2) the speed at which they can complete tasks, 3) the individual robot complexity, and 4) the complexity of the team. She also contributes to the development of a computational environment to simulate the performance of a team.

1.5 Thesis Organization

This thesis is divided into 5 chapters. Chapter 1 states the purpose and scope of this research and presents relevant background information. Chapter 2 develops the models that are used to generate an optimization problem. These models include a terrain model based on the statistical representation of different terrain features, a mobile robot model that provides an estimate of total robot mass, and a model for robot-terrain interactions on different types of terrain obstacles. Chapter 3 develops an optimization from the models, and describes a novel metric for measuring the performance of robot teams. It also presents methods to solve the optimization problem and to explore the design space of robot teams. Chapter 4 presents a case study of designing a team of robots for exploration on Mars. Finally, Chapter 5 presents a summary of the work in this thesis, along with conclusions and recommendations for future work.

Chapter 2

Modeling

This chapter develops several models that focus on the design of a mobile robot for operation in rough terrain. First, I present a model of a robot with a four-wheeled locomotion system and a model of terrain obstacles. This chapter next develops a model that determines a robot's capability to traverse the obstacles defined within the terrain model. Finally, I develop an appropriate system-level model of the robot that accounts for the effects of rough-terrain travel. Chapter 3 collects these models and develops an optimization problem to determine the most cost-effective robot design.

2.1 Robot Locomotion System

The robot locomotion system interacts with the terrain to generate motion for the robot. Numerous locomotion systems exist, most of which interact with the terrain through wheels, tracks, or legs. The Rocker-Bogie system has gained popularity through its use in NASA's Mars rovers.

I use a model of a simple four-wheeled robot as a generalized locomotion system. In practice, a designer could develop a model to account for the complexities of any locomotion system. Figure 2.1 shows the basic geometry of the robot. The robot's box-shaped body contains the subsystems of the robot as well as any additional payload. This model assumes that the robot's four cylindrical wheels are independently suspended and driven.

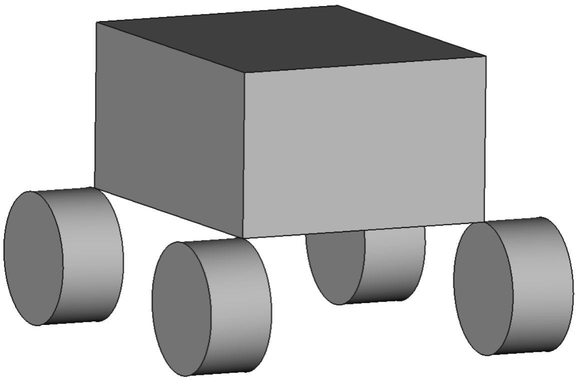


Figure 2.1: The simplified robot model used in analyzing terrain interaction mechanics

Figure 2.2 shows a profile view of the robot and indicates the dimensions relevant to locomotion. The height of the robot's body is based on the payload (such as tools or instruments) that the robot will carry, and is constant in this model. The robot's center of gravity (CoG) is centered vertically within the robot's body, but can be offset forwards or backwards. The clearance height of the robot's body is equal to the diameter of the wheels. Table 2.1 lists the variable robot dimensions in this model.

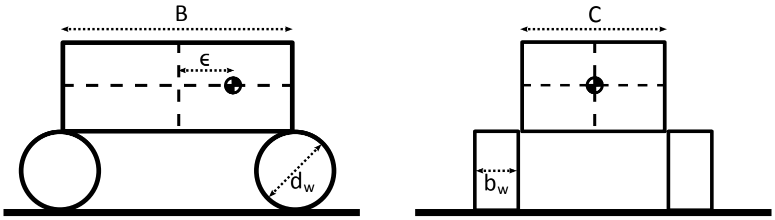


Figure 2.2: Front and side view of the robot marked with variable dimensions for the model

Table 2.1: Dimensions relevant to the locomotion system of the robot

Input	Symbol	Description
Wheelbase	B	Distance between the front and rear axles
Track	C	Distance between the inside of two wheels on the same axle
Wheel Diameter	d_w	The diameter of a wheel, not including the grousers
Wheel Width	b_w	The width of the wheel in contact with the ground
Center of Gravity (CoG) Bias	ϵ	The CoG bias is the forward distance from the geometric center of the body to the CoG. If the CoG is behind the geometric center, the CoG bias is negative.

2.2 Terrain Representation

The terrain upon which the robot operates is one of the most important considerations in the design of planetary robots. In order to accurately predict the performance of a mobile robot, the terrain must be characterized and modeled effectively. This section describes a model that characterizes the terrain obstacles into three categories, and statistically accounts for the uncertainty in the terrain.

2.2.1 Terrain Obstacle Characterization

"Terrain obstacle" refers to any landform or object on the ground that could impede the movement of a vehicle or robot across the terrain. The goal of characterizing terrain features into categories is to be able to accurately represent a region of land as a set of statistical models describing each obstacle category.

Figure 2.3 presents a simplified robot crossing two different terrain obstacles. The figure shows a robot climbing each obstacle at its threshold point: the point at which the robot experiences the maximum motion impedance from the obstacle. Because the forces experienced by the robot at the threshold point are the same for each obstacle, they could be categorized into the same group even if the two features seem different.

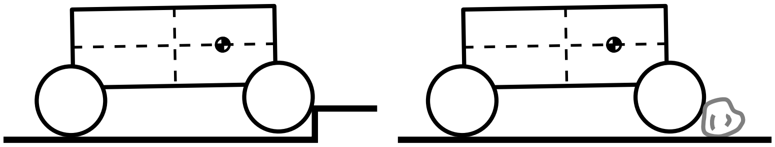


Figure 2.3: Demonstration of two terrain hazards with equivalent effects on robot mobility

There are several methods of categorizing terrain features. For example, the visual navigation system of the Mars Exploration Rovers discretizes terrain into cells and quantifies them in terms of three components: step, tilt, and roughness [44]. The step component accounts for differences in cell elevation compared to the average elevation of cells around it. The tilt component accounts for sloping terrain detected along a group of cells. The roughness component accounts for how much the elevation of individual cells in a group vary from a planar surface fit to the group. Note that this classification is used in a real-time navigation system, not for predicting robot performance during the design process.

It is important to note that the categorization of terrain features cannot perfectly represent every possible terrain condition encountered by a robot. I develop this model to provide reasonable obstacle representations, while requiring relatively straightforward calculations for robot performance.

The terrain model presented here considers three categories of obstacles to be representative of the majority of obstacles encountered. To simplify the problem, the model assumes two-dimensional interactions between robots and hazards. To categorize and quantify obstacles, this model first distinguishes three classes of terrain features based on size: features that are much smaller than the robot, features with sizes on the order of the robot size, features that are much larger than the robot.

Features that are much smaller than the robot, such as pebbles, will generate very little to no motion resistance and are excluded from the model.

Features that are much larger than the robot, such as craters or ridges, are also excluded from the model, since there will often be little uncertainty associated with them. Although larger features have the potential to have greater impacts on robot performance, their size allows them to be detected through satellite imaging or lidar. Because of this, they need not be accounted for in a model of the uncertainty in the terrain. Additionally, during the mission a robot would encounter significantly fewer of these obstacles than it would encounter smaller sized obstacles. These realizations make large features easier to account for through specific mission planning than through an uncertainty model.

Features with sizes on the order of the size of the robot are more common than larger obstacles and have significant impacts on robot performance. This model focuses on representing these obstacles, and first distinguishes them into grade-based obstacles and geometric obstacles.

Grade-based obstacles apply normal forces on the wheels that are perpendicular to the direction of travel for the robot. A simple case of this is sloped ground, like a ramp, demonstrated in Figure 2.4. In this model, grade-based hazards consist solely of sloped terrain, and will be referred to as "slopes." Slopes are considered to be continuously encountered, with horizontal ground considered to be a slope with zero angle.

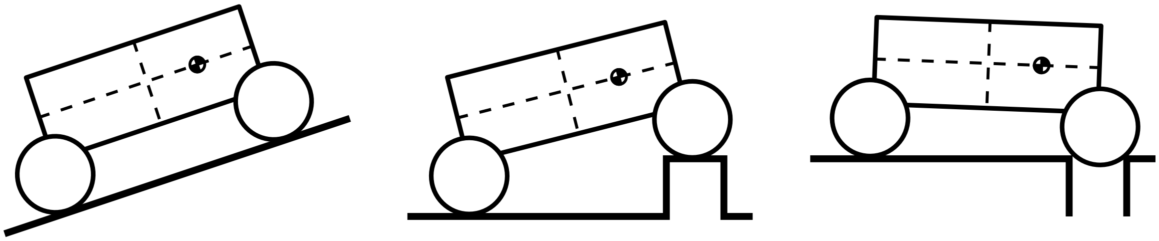


Figure 2.4: The three obstacle categories (from left to right) in the terrain model: slopes, bumps, and ditches

Geometric obstacles apply normal forces to the wheel that are not necessarily parallel to each other, and may be in or against the direction of travel. A simple example of this is a rock, which applies a force opposing the direction of travel. Though geometric hazards could take any shape, this model includes only box-shaped obstacles, called "bumps," and infinitely-deep holes, called "ditches" (see Figure 2.4). These general cases provide conservative estimates of the largest possible traversable obstacle.

Geometric obstacles of other shapes can be represented by a either bump or ditch with an equivalent size. Consider a two-dimensional interaction with any positive (above ground level) geometric hazard. The equivalent bump or ditch obstacle provides the same threshold impedance as the original hazard. In Figure 2.5, the threshold forces applied to the robot are the same for each obstacle.

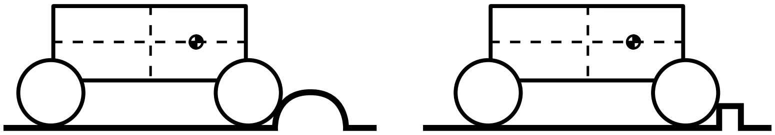


Figure 2.5: A round obstacle (left) represented by an equivalent bump obstacle (right)

The distinction in the model between positive obstacles (bumps) and negative obstacles (ditches) is to account for the different threshold impedances produced from the front wheels versus the back wheels of the robot climbing the obstacle. For bump obstacles, the normal reaction force on the non-climbing wheel is perpendicular to the direction of travel. For ditch obstacles, that force has a component in the direction of travel. Appendix B examines the mechanics of obstacle climbing and provides a more detailed explanation for this.

2.2.2 Slope Obstacle Representation

Slope obstacles are described by a single parameter: the angle of the slope, which is equal to the pitch angle imposed on the robot. While it's possible for the robot to be situated on wavy substrate, this model only considers slope obstacles on the order of the size of the robot (see Section 2.2.1). Slopes that change in shorter distances than the size of the robot do not necessarily pitch the robot at the angle of the slope (see Figure 2.6).

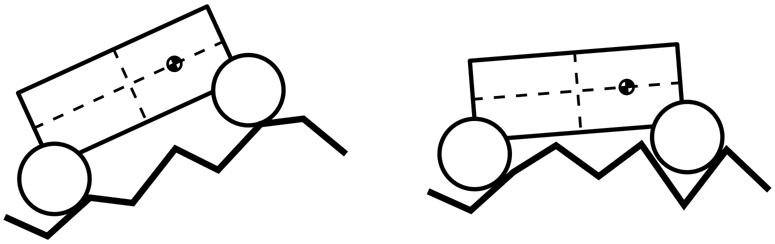


Figure 2.6: Steep slope changes on a scale smaller than the size of the vehicle may not significantly affect robot pitch

Additionally, the terrain model considers the robot to be constantly engaged with a slope obstacle, which includes a completely horizontal slope obstacle. This is beneficial for two reasons. First, the slope distribution can be easily determined from elevation data. In cases where the terrain is mostly flat, the slope distribution would be weighted very close to zero. Second, performance calculations are simplified, because slope obstacles are encountered at a known number of obstacles per distance traveled (see Section 3.2 for a description of the calculation). Equation 2.1 gives the approximate number of slope obstacles encountered by a robot with wheelbase B traveling a distance D.

$$N_{slopes} \approx \frac{D}{B} \tag{2.1}$$

Consider a robot placed into a terrain region at a random location and a random heading. The probability distribution function (PDF) for slopes describes the likelihood that the robot would be within a given range of slopes. A positive slope indicates the robot heading points uphill, and a negative slope indicates the robot heading points downhill.

2.2.3 Geometric Obstacle Representation

The terrain model describes both bumps and ditches by a single parameter, the "obstacle magnitude." For bumps, this parameter describes the height of the bump obstacle. For ditches, this parameter describes the length of the hole (the depth is assumed infinite). In both cases this is a conservative simplification, as I will explain below. Bumps and ditches are modeled using the same probabilistic representation, although the parameters in each distribution may differ.

A two-dimensional bump obstacle could be defined by two dimensions: a height and a length (Figure 2.7). However, the bump magnitude in the model only describes the height of the bump.

The length is assumed to be short enough so that the robot's front and rear wheels are never simultaneously climbing the obstacle. This assumption may often be untrue, but it is a conservative simplification. While a robot is climbing a bump obstacle, the maximum motion impedance is generated when the robot's front wheels are on the ground and its rear wheels are climbing the obstacle [45].

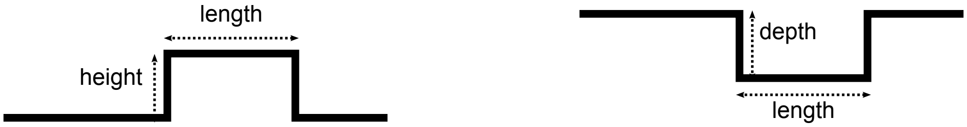


Figure 2.7: Bump and ditch obstacle with possible dimensions shown

A two-dimensional ditch obstacle (a negative hat obstacle), could also be defined by two dimensions, a depth and a length, as shown in Figure 2.7. The ditch magnitude in this model describes the length of the ditch, and the depth is assumed to be infinite. As with the bump, this is a conservative simplification because an infinite ditch results in the largest impedance forces. This is because an infinite ditch will allow the robot's wheel to drop down until it is in contact with both edges of the ditch. The floor of a finite ditch could catch the wheel before it makes contact with both edges. Additionally, an infinite ditch will immobilize a robot if its wheels cannot make contact with the other size of the ditch.

In the terrain model, geometric obstacle encounters are represented by a Poisson process. Their size is exponentially distributed. This is based on a statistical representation of rock sizes and abundances, referred to as a size-frequency distribution, for the Viking Lander 1 and Viking Lander 2 sites on Mars [46]. This representation considers rocks to be approximately spherical. Equation 2.2 shows the relationship developed by Golombek and Rapp, where N is the number of rocks per area with diameter h or greater. L is the total number of rock centers (for any size rocks) per area, and s is the parameter on the exponential rock size distribution. Equation 2.3 is the density function of the size-frequency distribution, related to N as shown in Equation 2.4. Figure 2.8 shows a comparison of the size-frequency density functions for geometric obstacles with different parameters.

$$N\left(h\right) = Le^{-sh} \tag{2.2}$$

$$n\left(h\right) = Lse^{-sh} \tag{2.3}$$

$$N\left(h\right) = \int\limits_{h}^{\infty} n\left(h\right) dh \tag{2.4}$$

The size-frequency distribution can be used to determine the number of obstacles (and their sizes) that a robot would encounter while driving along a known path length. The model assumes

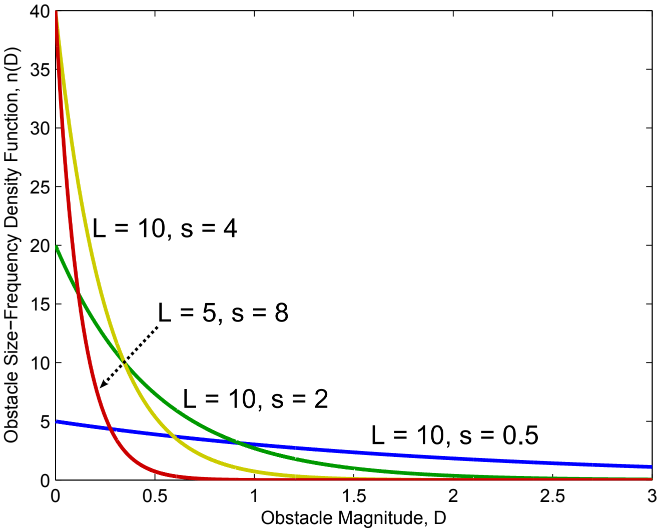


Figure 2.8: Size-frequency density functions with different parameters

that geometric-obstacles are roughly spherical in shape with their diameter equal to their magnitude. The frequency of obstacle encounters for a robot driving along a path is a Poisson process with rate parameter λ. $\lambda\left(y_0, y_1\right)$ is the average number of obstacles with diameter greater than y_0 and less than y_1 encountered by the robot per distance of travel.

λ can be determined by considering the path of the robot and the size-frequency distribution of rocks [41, 40]. Figure 2.9 shows the area swept out for a robot with track C while traveling a distance, D. The robot will encounter obstacles with size D if their centers are within the dotted lines. Note that the distance between the dotted lines is different for every obstacle size.

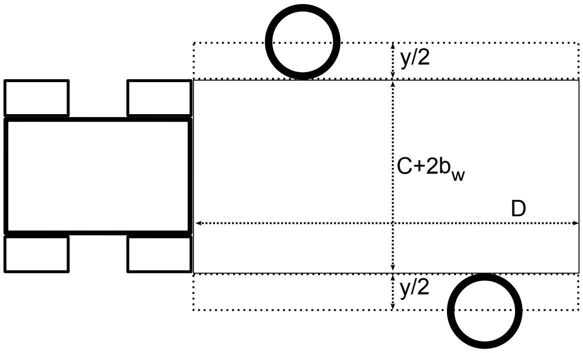


Figure 2.9: Path width for obstacle encounters

Equation 2.5 gives $\lambda\left(y_0, y_1\right)$ by integrating over the range of obstacle sizes and their related path widths.

$$
\begin{aligned}
\lambda(y_0, y_1) &= \int\limits_{y_0}^{y_1} (C + 2b_w + y)\, n\,(y)\, dy \\
&= \frac{L}{s} \left[e^{-sy_0} \left(sy_0 + s\,(C + 2b_w) + 1 \right) - e^{-sy_1} \left(sy_1 + s\,(C + 2b_w) + 1 \right) \right] \qquad (2.5)
\end{aligned}
$$

A similar method can be used to determine the average size of the obstacles that are encountered using the PDF of obstacle sizes in Equation 2.6

$$
f\,(y) = se^{-sy} \qquad (2.6)
$$

2.3 Robot-Terrain Interactions

This section develops models for the interactions between the robot and the three categories of terrain obstacles in the terrain model. I first develop a model of the forces generated between the wheel and the soil based on terramechanics. The robot-obstacle interaction models analyze the forces generated while the robot is traversing an obstacle. Together, these models can determine whether the robot can generate enough traction to drive over a given obstacle.

The models in this section make two notable approximations. First, quasi-static approximations are used for robot-obstacle interactions. At low speed this is a reasonable approximation. If desired or if necessary, a designer could implement a dynamic analysis to achieve improved accuracy. Second, robot-obstacle interactions are approximated with a two-dimensional analysis. Again, a designer could use a three-dimensional analysis. A two-dimensional model provides a conservative estimate, since in reality many terrain obstacles will only affect one wheel at a time.

2.3.1 Wheel-Soil Interaction

The wheel-soil interaction model examines the physical interactions between the wheel and the soil in order to quantify various forces that are produced. This model determines the maximum forward force, referred to as the drawbar pull, that the robot can generate. The total drawbar pull is the sum of the individual drawbar pull contribution from each wheel. This involves determining components of both motion resistance and forward thrust generated between the wheel and the ground. The magnitude of the components depend on different properties of the soil as well as the dimensions of the wheels and the loading on the wheels.

This model is based on terramechanics and primarily utilizes Bekker theory to model the physical interactions between the wheel and the soil [26, 47, 27, 30]. Specifically, it uses the relationships for a rigid wheel in loose soil. Bekker theory builds off of the soil mechanics work by Terzaghi, who developed empirical relationships to determine soil parameters used in Bekker's relationships [28, 29].

I separate the components of the drawbar pull contribution at each wheel into two tractive components and three resistive components, shown in Equation 2.7. An important distinction between tractive and resistive forces is that tractive forces put energy into the motion of the robot while resistive forces put work into tearing or reshaping the substrate. The tractive components of drawbar pull include soil thrust and grouser thrust; the resistive components include soil compaction, bulldozing, and rolling resistance. All five components are listed and described in Table 2.2. Appendix A presents further explanation of the components of drawbar pull, along with their detailed calculations.

$$F_{dp} = F_s + F_g - R_c - R_b - R_r \tag{2.7}$$

Table 2.2: Components of the drawbar pull contribution of a wheel

Component of Drawbar Pull	Symbol	Description
Soil Thrust	F_s	Tractive force generated by cohesive forces within the soil that resist deformation
Grouser Thrust	F_g	Tractive force generated by the motion of the of grousers (or “lugs”) through the soil
Compaction Resistance	R_c	Resistive force originating from the internal resistance of the soil to being depressed by loading of the wheel
Bulldozing Resistance	R_b	Resistive force from the wheel pushing soil material forward and up along the side of the wheel, rather than pulling it backward directly underneath the wheel
Rolling Resistance	R_r	A combination of resistive forces internal to the wheel, including friction and wheel deformation

2.3.2 Slope Obstacle Interaction

The slope interaction model considers three generalized cases of slope traverses: uphill climbs, downhill climbs, and crosshill climbs (see Figure 2.10). During uphill and downhill climbs, the robot only experiences rotation along the pitch axis. During crosshill climbs the robot only experiences rotation along the roll axis. In reality the robot will be traversing slopes that cause simultaneous rotation along both axes. Because the uphill/downhill and crosshill components are non-additive, the robot can (based on this quasi-static analysis) experience rotations up to the maximum stable slope angle along each direction while still remaining stable.

The slope interaction model determines whether terrain with a given slope can be traversed by the robot based on two factors: stability and traction. Stability requires that the robot does not pitch forward or backwards onto its back, thus becoming immobilized. Traction requires that the robot can generate enough force at the wheel-soil interface to maintain speed on the slope.

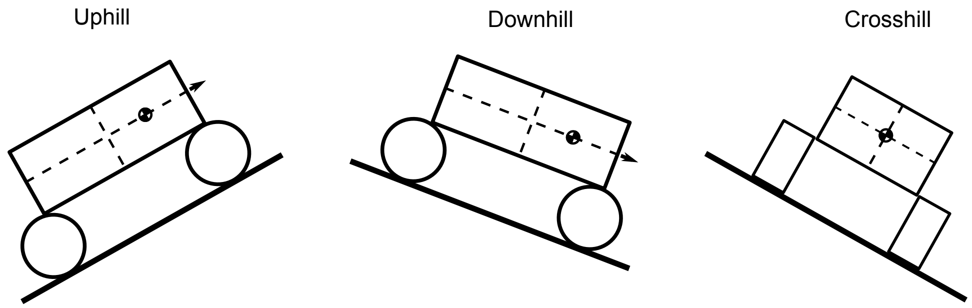


Figure 2.10: Analyses for the robot-slope interaction model considers uphill, downhill, and crosshill travel

Appendix B.1 covers a detailed stability and traction analyses for the three categories of slopes.

In practice, driving the robot across any slopes that are close to the maximum stable slope is dangerous. Any perturbations or effects of dynamics could cause the robot to rotate beyond its stable limits. Additionally, driving on a slope where traction cannot be maintained could cause problems with the robot's navigation. If desired, one could add a margin of safety to the maximum stable uphill, downhill, and crosshill slopes.

2.3.3 Geometric Obstacle Interaction

The geometric interaction model determines if a bump or ditch obstacle of a given magnitude can be successfully traversed by the robot based on three factors: stability, traction, and geometry. Figures 2.11 and 2.12 illustrate these limitations for bumps and ditches. Stability requires that the robot does not pitch forward or backwards onto its back, thus becoming immobilized. Traction requires that the robot can generate enough force at the wheel-soil interface to surmount the obstacle. Geometry requires that the robot is not blocked or immobilized due to the body of the robot colliding with the obstacle's surface. These three factors are investigated for the robot crossing the obstacle with both its front wheels and its rear wheels. Appendix B.2 provides the full details of this analysis for bump obstacles, and Appendix B.3 provides the analysis for ditches.

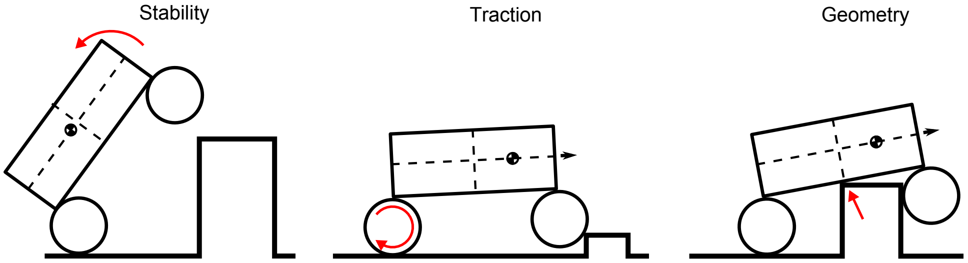


Figure 2.11: Stability, traction, and geometric concerns for a robot driving over a bump

The geometric obstacle interaction models are based on two-dimensional, quasi-static analyses. For a given obstacle size, the analysis checks whether a robot can pass the stability and geometry tests, and determines the traction that the robot would need to generate to drive over the obstacle.

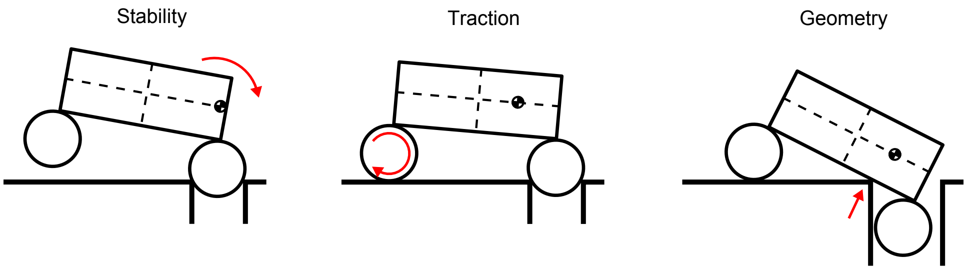


Figure 2.12: Stability, traction, and geometric concerns for a robot driving over a ditch

The wheel-soil interaction model can then be used to determine whether or not the robot is capable of generating the required traction.

In the case of the ditch obstacle, under almost all circumstances it would be impossible to have a robot that passes the geometry requirements but fails the stability requirements. Because of this, the ditch analysis excludes the stability requirement.

2.4 Robot Subsystems

In the context of planetary exploration, robots receive commands, collect information, make decisions, and physically interact with their environment. This section develops simplified models of robot subsystems, making reasonable assumptions with respect to their designs. The purpose of these subsystem models is to estimate masses of the subsystems based on their requirements. Table 2.3 presents a description of the robot subsystems.

The robot, shown in Figure 2.1, consists of a box-shaped body and four wheels that extend from the body. The wheels and body are connected via a ladder-shaped chassis. The body contains the robot subsystems, including the electronics, instruments and tools, thermal management hardware, and battery. The robot is primarily powered by body-mounted solar panels, with batteries as a source of backup power. Hub motors in each wheel drive the robot.

The payload, thermal management, and electronics subsystems are not explicitly analyzed. Instead, the model assumes that they each contribute a constant, known amount to the overall mass of the robot. The variation in the mass of these systems across robot designs is likely to only make a small contribution relative to the mass of the other systems.

The chassis, motors, and power subsystems are sized to meet the robot requirements based on the terrain conditions. The chassis is sized to be strong enough to withstand the stresses induced by the robot driving over the largest sized obstacles that it can traverse. Appendix C presents analyses to determine chassis stresses induced by different obstacles. The motors and power subsystems are sized based on the average power that the robot consumes.

Table 2.3: Descriptions of the subsystems in the robot model

Subsystem	Description
Chassis	The chassis is modeled as a ladder frame with a rectangular cross section. Both vertical and horizontal thicknesses are consistent across all rungs. The strength of the chassis is determined by the maximum stress imposed by terrain hazards.
Payload	The payload includes any tools, instruments, or scientific equipment that the robot will carry.
Motors	Hub motors in each wheel drive and steer the robot. The size of the motors is determined by the average drive power of the robot, which depends on the terrain conditions.
Wheels	The wheels are rigid (compared to a flexible rubber tire) and have grousers to improve performance.
Electronics	This includes sensors, computers, communication devices, and other necessary electronics to operate the robot. Notable exclusions are non-essential electronics included with the payload equipment and power management electronics.
Thermal Management	The thermal management system includes any necessary cooling or heating devices to maintain the robot's temperature inside the operating range for the hardware.
Power	The power supply subsystem consists of solar panels, batteries, and the power management hardware. The solar panels are sized to meet the average power requirements for the robot. The batteries, which provide supplemental power during periods of low sunlight, are sized to provide full backup power for a set period of time. The power management hardware includes the electronics required to transport and convert power to different pieces of hardware on the robot.

2.4.1 Power Usage

This model considers that the robot consumes a constant amount of energy for its electronics plus additional energy for traveling. The average power consumed during travel, P_{drive}, is determined from the robot capabilities and the terrain model, as shown in Equation 2.8. R is the average resistive force from slopes, bumps, or ditches. $\dot{N}$ is the rate (number per distance) of obstacles encountered. $\bar{Y}$ is the average magnitude of bumps or ditches that are traversed. $\dot{N}\bar{Y}$ estimates the percentage of the time that the robot is in contact with that type of obstacle. The robot constantly experiences the resistive force from slopes, R_{slope}, which includes resistive forces from the soil mechanics.

$$P_{drive} = v\left(R_{slope} + \dot{N}_{bump}\bar{Y}_{bump}R_{bump} + \dot{N}_{ditch}\bar{Y}_{ditch}R_{ditch}\right) \tag{2.8}$$

Equation 2.9 shows that average resistive force, R, is estimated as 50 percent of the threshold impedance, the average of the front and rear threshold impedances, from a mean-sized obstacle. Note that this is the mean obstacle size that the robot can traverse, not the mean size of all obstacles.

$$R_{obstacle} = \frac{1}{2}\frac{R_{rear} + R_{front}}{2} \tag{2.9}$$

Equation 2.10 gives the average angle of slopes that the robot drives on, based on the probability density function (PDF) of slopes, $f_\theta(\theta)$. This only includes slopes that the robot can safely drive on. θ_{up} and θ_{dn} are the angles of the maximum uphill and downhill slopes that the robot can traverse. The resistance of the slope is determined using the analysis in Section 2.3.2.

$$\bar{\theta} = \frac{\int_{\theta_{dn}}^{\theta_{up}} \theta f_\theta(\theta)\, d\theta}{\int_{\theta_{dn}}^{\theta_{up}} f_\theta(\theta)\, d\theta} \tag{2.10}$$

Equation 2.11 gives the number of bump and ditch obstacle encounters per distance traveled based on the obstacles' size-frequency distributions, $n_y(y)$, and the maximum traversable obstacle magnitude, $y^\star$. Note that $\dot{N}$ is equal to , λ, the obstacles encountered per driving distance determined in Equation 2.5.

$$\dot{N} = \int_0^{y^\star} (C + 2b_w + y)\, n_y(y)\, dy \tag{2.11}$$

Equation 2.12 gives the mean magnitude of obstacles that are traversed by the robot. $f_y(y)$ is the PDF of obstacle sizes, y (see Equation 2.6). The resistance of the average obstacle (bump or ditch) can be determined using the analyses in Appendix B.

$$\bar{Y} = \frac{\int_0^{y^\star} y\,(C + 2b_w + y)\, f_y(y)\, dy}{\int_0^{y^\star} (C + 2b_w + y)\, f_y(y)\, dy} \tag{2.12}$$

2.4.2 Robot Mass

The total robot mass is the sum of the masses of the subsystems (Equation 2.13). The mass of the team is the number of robots times the mass of one robot (Equation 2.14). This section describes how the model determines the mass of each subsystem.

$$m_{robot} = m_{payload} + m_{electronics} + m_{thermal} + m_{chassis} + m_{motors} + m_{wheels} + m_{power} \tag{2.13}$$

$$m_{team} = r m_{robot} \tag{2.14}$$

Payload, Electronics, and Thermal Subsystems The robot model assumes that the payload, electronics, and thermal subsystems have a known, constant mass regardless of the robot design. For the payload system this is a valid assumption, since the equipment required for the mission tasks should not depend on the robot design. The electronics should also be similar across robot designs. For simplicity the model assumes that the thermal management subsystem has a set mass. Though the design and mass of the thermal management system would change across robot designs, the variation will likely be small relative to the total robot mass.

Chassis The mass of the chassis, shown in Figure 2.13, can be determined from the material properties and dimensions of the chassis. Equation 2.15 gives the mass of the chassis. ρ is the density of the chassis material, b_c and h_c are the width and height of the chassis cross section.

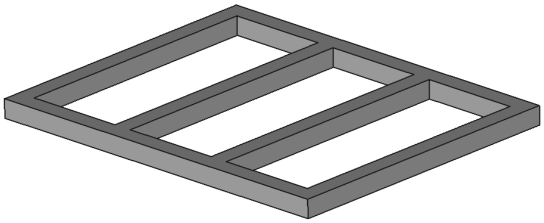


Figure 2.13: The chassis of the robot, modeled as a ladder frame with four rungs

$$m_{chassis} = \rho b_c h_c \left(2B + 4C\right) \tag{2.15}$$

Motors The mass of each motor is determined from its power requirement, P_{drive}, determined in Section2.4.1. based on an exponential relationship, shown in Equation 2.16. Curve fitting based on data from commercially available motors, shown in Figure 2.14, results in parameters of $a = 9.67$ and $b = 0.61$.

$$m_{motors} = 4a \left(\frac{P_{drive}}{4}\right)^{b} \tag{2.16}$$

Wheels The mass of wheels depends on their dimensions. The wheel mass is determined from an empirical model presented by Alibay et al [12]. Equation 2.17 shows a modified version of the relationship that depends on wheel diameter, d_w, and wheel width, w_w. Dimensions are in meters and mass is in kilograms.

$$m_{wheels} = 4 \times 26.5 \left(d_w\right)^{2.81} \left(b_w\right)^{0.4} \tag{2.17}$$

Power Supply The mass of the power supply system includes the mass of the solar panels, batteries, and power management electronics:

$$m_{power} = m_{solar} + m_{battery} + m_{management} \tag{2.18}$$

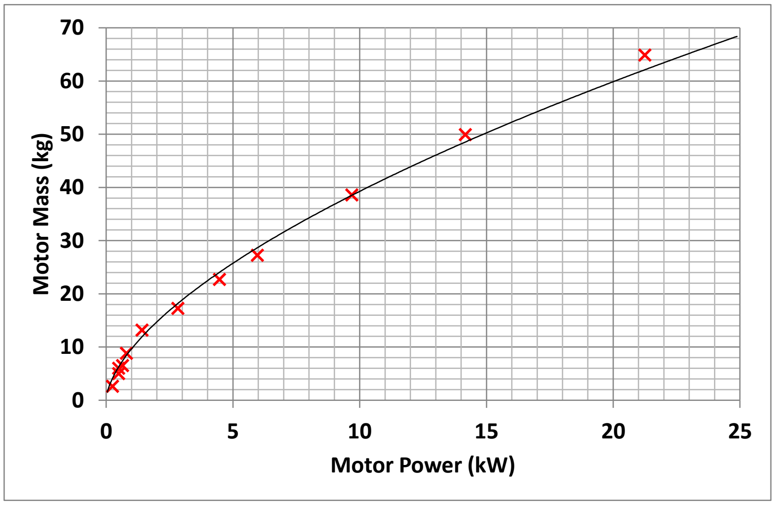


Figure 2.14: Empirical motor power-to-mass relationship based on electric vehicle motors

The solar panels are sized to meet the average power used by the robot. Equation 2.19 shows how the model determines the mass of the solar panel based on the ρ_{solar}, the density of the solar panel in mass per area; P, the average robot power required; s, an oversizing factor; η, the efficiency of the solar panels; and I, the solar irradiance (power per surface area).

$$m_{solar} = \frac{\rho_{solar} P s}{\eta I} \tag{2.19}$$

The battery is sized to provide full backup power to the robot for a certain period of time. Equation 2.20 shows the mass of the battery, which depends on T_{backup}, the desired period of time to have full backup power; P, the average robot power; and c_{batt}, the specific energy of the battery chemistry.

$$M_{battery} = \frac{T_{backup} P}{c_{batt}} \tag{2.20}$$

The model assumes that the mass of the power management electronics are linearly related to the average power of the robot, as show in Equation 2.21. k is the linear coefficient in units of mass per power.

$$M_{management} = k_P P \tag{2.21}$$

Chapter 3

Optimization

This chapter presents the formulation of an optimization problem from the models developed in Chapter 2. The solution to this optimization problem provides insights into the optimal degree of fractionation for a team of robots. Section 3.1 describes the design variables and design constraints chosen for the optimization. Section 3.2 develops a measure of performance for a team of robots, which is the objective function for the optimization. Section 3.3 presents a computer program to computationally solve the optimization. Additionally, the program includes a graphical user interface that is useful for tradespace exploration and design comparisons.

3.1 Optimization Problem Formulation

The design framework uses the models developed in Chapter 2 to develop an optimization problem that provides the optimal robot team design for given terrain conditions. In practice, it is possible to run one optimization to determine the optimal number of robots in the team. However, in most cases the models will lack sufficient completeness for the optimization to give a true optimum. In those cases, the optimization problem would serve as a tool to explore the design tradespace, allowing a designer to investigate the sensitivity of the team design to changes in terrain conditions. The optimization can be used to determine a range of team sizes that provide an acceptable level of performance. The decision of the number of robots to use for the mission can depend on the results of several optimizations, as well as additional factors that may be too complex to model.

Optimization algorithms typically accept optimization problems that are written in the common form shown in Equation 3.1. $f(x_i)$, known as the "objective function," is the scalar quantity to be either maximized or minimized (see Section 3.2). x_i, known as the "decision variables" (DVs), are parameters that are adjusted in order to optimize the objective function. $\vec{c}(x_i) \leq 0$ is a set of equations, known as "constraints," that restrict the possible values of the DVs.

$$\min_{x_i} f(x_i) \quad (3.1)$$

$$\text{such that}\, \vec{c}(x_i) \leq 0$$

Table 3.1 lists the DVs for the optimization. Refer to Section 2.1 and Table 2.1 for more information about the decision variables related to the robot model. Refer to Appendix B and Appendix C to see how the models depend on the obstacle decision variables.

Table 3.1: Decision variables for the optimization

Name	Symbol	Description
Wheelbase and Track	B, C	The wheelbase and track, which are assumed equal, for every robot
Wheel Diameter	d_w	The diameter of the robot's wheels
Wheel Width	b_w	The width of the robot's wheels
CoG Front Bias	ϵ	The distance from the geometric center of the body to the CoG
Chassis Vertical Thickness	h_c	The height of a cross-sectional slice of a beam in the ladder frame
Chassis Horizontal Thickness	b_c	The width of a cross-sectional slice of a beam in the ladder frame
Allowable Uphill Slope	θ_{up}	The steepest uphill slope that the robot will traverse
Allowable Downhill Slope	θ_{dn}	The steepest downhill slope that the robot will traverse
Allowable Positive Obstacle	h	The largest positive obstacle that the robot will traverse
Allowable Negative Obstacle	L	The largest negative obstacle that the robot will traverse

The decision variables may seem redundant, since the traversable obstacle magnitudes are completely dependent on the robot design DVs. In other words, the maximum slopes and obstacle magnitudes for a robot should be determinable from the robot design and the terrain parameters. The reason for using these DVs is that some of the relationships in the models are difficult or impossible to solve. For example, the terramechanics model (see Section 2.3.1) used in the obstacle models cannot be back-solved to determine the inputs for given outputs. By including the obstacle magnitudes in the DVs, they are effectively solved by the optimization routine.

Table 3.2 lists the optimization constraints, along with relevant relationships from the models. Note that the constraint expressions are limited to less than or equal to zero.

Table 3.2: Optimization constraints

Constraint Name	Constraint Expression (must be ≤ 0)	Equations of Origin	Description
Mass	$m_{team} - m_{budget}$	2.14	Limits the team mass to be within the mass budget, m_{budget}
Uphill Stability	$\theta_{up} - \theta_{up}^{stable}$	B.1	Ensures that the allowable uphill slope, θ_{up}, does not cause the robot to tip over
Uphill Traction	$-F_{dp,up}$	B.6	Ensures that the robot can generate enough traction on the allowable uphill slope, θ_{up}
Downhill Stability	$\theta_{dn} - \theta_{dn}^{stable}$	B.2	Ensures that the allowable downhill slope, θ_{dn}, does not cause the robot to tip over
Crosshill Stability	$\frac{\theta_{up}-\theta_{dn}}{2} - \theta_{cs}^{stable}$	B.3	Ensures that the robot is stable on a crosshill slope approximated by the mean of the allowable uphill and downhill slopes
Bump Front Traction	$F_{T,2} - F_{T,2}^{terra}$	B.15	Ensures that the robot can generate enough traction to cross the maximum allowable positive obstacle, h, with its front wheels. $F_{T,2}$ is the required traction and $F_{T,2}^{terra}$ is the available traction.
Bump Rear Traction	$F_{T,1} - F_{T,1}^{trac}$	B.26	Ensures that the robot can generate enough traction to cross the maximum allowable positive obstacle, h, with its rear wheels. $F_{T,1}$ is the required traction and $F_{T,1}^{terra}$ is the available traction.
Bump Force	$-R_2$	B.25	Ensures that the reaction force on the rear wheel is not negative (which is not physically possible). It is possible for the other bump constraints to be met otherwise, which would falsely indicate that the robot can traverse an obstacle.
Bump Stability	$h - h_{max}$	B.8, B.9	Ensures that the robot is stable while crossing the maximum allowable positive obstacle, h

Constraint Name	Constraint Expression (must be ≤ 0)	Equations of Origin	Description
Bump Clearance	$h - d_w$	B.30	Ensures that the robot does not high-center on the maximum allowable positive obstacle, h. Recall the d_w is also the clearance height of the robot's body.
Ditch Front Traction	$F_{T,2} - F_{T,2}^{terra}$	B.36	Ensures that the robot can generate enough traction to cross the maximum allowable negative obstacle, L, with its front wheels. $F_{T,2}$ is the required traction and $F_{T,2}^{terra}$ is the available traction.
Ditch Rear Traction	$F_{T,1} - F_{T,1}^{terra}$	B.49	Ensures that the robot can generate enough traction to cross the maximum allowable negative obstacle, L, with its rear wheels. $F_{T,1}$ is the required traction and $F_{T,1}^{terra}$ is the available traction.
Ditch Clearance	$d_w - L$	B.54	Ensures that the robot's wheels are large enough to prevent the robot from falling into the maximum allowable negative obstacle, L
Uphill Stress	$\sigma_{up} - \sigma_{yield}$	C.5, C.6, C.10	Ensures that the robot's chassis is strong enough to travel on the maximum allowable uphill slope, θ_{up}
Downhill Stress	$\sigma_{dn} - \sigma_{yield}$		Ensures that the robot's chassis is strong enough to travel on the maximum allowable downhill slope, θ_{down}
Crosshill Stress	$\sigma_{cs} - \sigma_{yield}$	C.16	Ensures that the robot's chassis is strong enough to travel on a crosshill slope approximated by the mean of the allowable uphill and downhill slopes
Bump Stress	$\sigma_{bump} - \sigma_{yield}$	C.21 to C.26	Ensures that the robot's chassis is strong enough to traverse the maximum allowable positive obstacle, h

Constraint Name	Constraint Expression (must be ≤ 0)	Equations of Origin	Description
Ditch Stress	$\sigma_{ditch} - \sigma_{yield}$	C.27 to C.32	Ensures that the robot's chassis is strong enough to traverse the maximum allowable negative obstacle, L
Wheel Separation	$d_w - B + s_w$		Limits the size of the wheels so that the front and rear wheel are not in contact. The minimum allowed separation between the wheels is s_w.
CoG Forward Limit	$\epsilon - \frac{B}{2}$		Limits the longitudinal location of the CoG to within the body of the vehicle
CoG Backward Limit	$-\epsilon - \frac{B}{2}$		
Minimum Wheel Width	$r_w - \frac{b_w}{d_w}$		Requires the width of the wheel to be at least a certain fraction of the diameter, as specified by r_w
Chassis Thickness Ratio	$\frac{h_c}{b_c} - r_t$		Requires the horizontal thickness of the chassis cross section to be at least a certain fraction of the vertical thickness, as specified by r_t
Sinkage Limit	$z_w - r_s d_w$	A.5	Limits the amount that the wheels of the robot can sink into the soil. r_s is the upper limit of the ratio of sinkage to wheel diameter.

3.2 Performance Metric

A challenge to mobile robot design is determining an effective method to evaluate robot designs. Researchers have developed several different metrics for mobile robot performance (see Section 1.4.3). These metrics are very useful for evaluating the relative mobility characteristics of robot designs. However, they do not provide a relatable indication of robot performance, because they do not consider the terrain conditions. Additionally, the mobility metrics are targeted at single robots, rather than teams.

In this section I develop a performance metric for single mobile robots referred to as the "equivalent straight-line velocity" (ESLV), and a related performance metric for teams of mobile robots referred to as the "combined equivalent straight-line velocity" (CESLV). Both metrics provide relatable measures of mission performance.

3.2.1 Time Delays from Obstacles

To calculate ESLV or CESLV, the obstacles in the terrain model (Section 2.2) need to be related to the robot performance. Specifically, it requires a model that relates the magnitude of each obstacle type to the time it would take the robot to either drive over the obstacle or drive around the obstacle. Developing a model to do this is difficult because it is dependent on many factors, each of which could significantly affect the result. For the purposes of demonstration, I present simplistic linear models that can be used to determine the time delays for the robot to overcome each obstacle type. Note that the time delay is not the total time for the robot to traverse or maneuver around the obstacle, but is the difference between that time and the time to travel along the same length of level, unhindered terrain.

For positive and negative obstacles (bumps and ditches) that the robot is capable of traversing, the time delay is based on the maximum obstacle size of that type that the robot can traverse:

$$\hat{t}_{bc} = \hat{t}_{dc} = \frac{t^\star}{y^\star} y \tag{3.2}$$

$t^\star$ is the time delay for traversing an obstacle of magnitude $y^\star$. $y^\star$ is the maximum obstacle magnitude that the robot can traverse. y is the magnitude of the obstacle of interest.

For positive and negative obstacles that the robot cannot traverse, the time delay is based on the time to drive around that obstacle. The actual path length added by a robot maneuvering around an obstacle depends on several factors specific to the robot. For example, if the robot can detect the obstacle far in advance, it can drive in a smooth path around the obstacle, reducing the extra distance it has to drive. The extra distance also depends on where the robot encounters the obstacle along its front face. If the robot encounters the obstacle dead-center, it would have to drive further than if it encountered it only along the outside of one wheel.

Figure 3.1 shows one approximation of the path distance for the robot to drive around an obstacle, assuming it has a circular profile. As a simple approximation, the total distance to drive around an obstacle of magnitude y is $\frac{\pi y}{2}$, or half of the perimeter around the obstacle (see Figure 3.1). Compared to the actual path that the center of the robot would trace around the obstacle, this approximation reduces the complexity of the calculations in the following section. The unhindered distance would be y, so the additional distance is

$$\hat{d}_{bm} = \hat{d}_{dm} = \frac{\pi y}{2} - y = \frac{(\pi - 2)\, y}{2} \tag{3.3}$$

With a nominal robot velocity of v, the time penalty is

$$\hat{t}_{bm} = \hat{t}_{dm} = \frac{\hat{y}}{v} = \frac{(\pi - 2)\, y}{2v} \tag{3.4}$$

For uphill slopes that the robot can traverse, the time delay is approximately proportional to the resistive force of gravity. For a slope of angle θ, the component of gravity acting against the motion of the robot is $\sin\theta$. The time delay is also dependent upon the maximum uphill slope

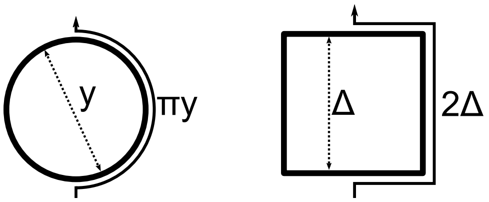


Figure 3.1: Distances to drive around untraversable bump and ditch obstacles (left) and untraversable slope obstacles (right)

that the robot can climb, the size of the slope obstacle, Δ, as well as a dimensionless parameter k. Applying the small angle approximation ($\sin\theta \approx \theta$) gives

$$\hat{t}_{usc} = \frac{k\Delta}{v\theta_{uphill,max}}\theta \tag{3.5}$$

As discussed in Section 2.2.2, the size of the slope obstacle is equal to the wheelbase and track of the robot.

The time delay for traversable downhill slopes is zero (Equation 3.6). This model assumes that the speed of a robot is limited by its navigation capabilities, so the robot cannot travel faster downhill.

$$\hat{t}_{dsc} = 0 \tag{3.6}$$

For uphill or downhill slopes that cannot be traversed, the time delay is based on the distance to drive around the slope obstacle. This is similar to untraversable obstacles, except that the slope obstacle is square with size Δ:

$$\hat{t}_{usm,dsm} = \frac{\Delta}{v} \tag{3.7}$$

3.2.2 Equivalent Straight-Line Velocity (ESLV)

ESLV is a measure of the speed at which a robot can travel from one location to another, based on statistics for the obstacles in the region of travel. ESLV, v_{esl}, is equal to the straight-line distance, D, between two locations divided by the mean predicted time, $E[T]$, for the robot to travel between the two locations. The time to travel between the locations depends on the number and magnitude of the obstacles encountered, so ESLV will always be slower than the nominal robot velocity (traveling on level, flat soil).

$$v_{esl} = \frac{D}{E[T]} \tag{3.8}$$

Assume that every robot design would have similar abilities in terms of performing the necessary mission tasks while at any location of interest. (If the task-related capabilities of the robots would change across designs, that effect could be included in the robot model and supplemented to the

ESLV.) Regardless of the mission plan, the robot with the fastest ESLV will be able to travel fastest and thus perform the mission the fastest on average. If the subsequent target locations depend on the results of tasks from previous locations, the robot with the highest ESLV will be the quickest to reach the next target location once it is chosen.

In additional to being an effective metric for comparison during robot design, ESLV provides practical information regarding mission performance. Once ESLV is determined, a mission planner can use ESLV to estimate the average time it would take for a robot to visit a series of waypoints, as shown in Equation 3.9.

$$E\left[T\right] = \frac{D}{v_{esl}} \tag{3.9}$$

To determine ESLV, calculate T as the base time, $\bar{T}$, plus the mean additional time (referred to as "time delays") spent traversing or maneuvering around obstacles, $\hat{T}$:

$$T = \bar{T} + \hat{T}_{tot} \tag{3.10}$$

$\bar{T}$ is the robot's travel time if there were no obstacles (slopes, bumps, or ditches). Shown in Equation 3.11, it is equal to the distance traveled divided by the nominal robot velocity, v:

$$\bar{T} = \frac{D}{v} \tag{3.11}$$

The mean total time delay, $\hat{T}_{tot}$, is equal to the sum of the mean time delays from each obstacle encountered by the robot (Equation 3.12). This sum can be divided into eight sums, as shown in Equation 3.12, based on eight obstacle classes from the terrain model presented in Section 2.2. There are four pairs of sums for the four obstacle types (downhill and uphill slopes are considered separately), including one sum for traversable obstacles and one sum for untraversable obstacles of each type. This separation is necessary because the time delays are different for traversable and untraversable obstacles. Table 3.3 lists the subscript nomenclature.

$$\hat{T}_{tot} = \sum_{obstacles} \hat{T} = \hat{T}_{bc,tot} + \hat{T}_{bm,tot} + \hat{T}_{dc,tot} + \hat{T}_{dm,tot} + \hat{T}_{usc,tot} + \hat{T}_{usm,tot} + \hat{T}_{dsc,tot} + \hat{T}_{dsm,tot} \tag{3.12}$$

Table 3.3: Subscript nomenclature for the mean time delays in the ESLV calculations

Obstacle	Time delay due to:	
	Traversing the obstacle	Maneuvering around the obstacle
Bumps	*bc*	*bm*
Ditches	*dc*	*dm*
Uphill Slopes	*usc*	*usm*
Downhill Slopes	*dsc*	*dsm*

Because each set of obstacles are independent and identically distributed, the sums are equal to the products of the mean number of obstacles encounters and the mean time delay per encounter:

$$\hat{T}_{obs,tot} = N_{obs}\hat{T}_{obs} \tag{3.13}$$

obs indicates that the quantity is for any one of the eight obstacle classes. The models from Section 2.2 provide the necessary information to calculate the twelve quantities (eight N values and eight $\hat{T}$ values) needed to determine $\hat{T}_{tot}$.

Slopes follow a Gaussian distribution, so the average slope in each class can be determined by integrating the marginal PDF for that class. This is the same method used to determine the robot's driving power in Section 2.4.1. The bounds on Equation 2.10 simply need to be modified to the range of interest. Because the time penalties scale linearly with the obstacle magnitude, the mean obstacle size can be used in the time delay relationships from Section 3.2.1.

The number of bumps or ditches encountered can be determined from the average obstacle encounter rate. Equation 2.5 gives the average rate (obstacles per distance) of obstacle encounters for a specified range of obstacles. Multiplying the rate by the travel distance gives the mean number of obstacles encountered.

The mean number of uphill and downhill slopes can be determined using the PDF of slope angles with Equation 2.1. Equation 3.14 shows that the mean number of slopes is equal to the total number of slopes times the integral of the slope PDF over the slope angles of interest.

$$N_{slopes} = \frac{D}{B}\int f_{\theta}\left(\theta\right) d\theta \tag{3.14}$$

Table 3.4 presents the mean number of obstacle encounters and the mean time penalty for all eight classes of obstacles. σ, L, and s are parameters from the terrain model in Section 2.2. D is the path length, and $y^{\star}$ is the maximum traversable obstacle magnitude for either bumps or ditches. These expressions can be used in Equations 3.12 and 3.13 to determine to total delay time.

However, the time penalty given by Equation 3.12 is for a path length of D. As the robot encounters untraversable obstacles, the total distance it travels increases, which increases the number of obstacles it could encounter. Appendix D describes how to account for this effect in ESLV via an infinite geometric series. The resulting relation for ESLV is Equation 3.15, with K_c and K_m given by Equations 3.16 and 3.17. Note that the final value of v_{esl} is independent of D, the distance between the start and end point.

$$v_{esl} = v\frac{1 - K_m}{1 + K_c} \tag{3.15}$$

$$K_c = \frac{N_{usc}\hat{T}_{usc} + N_{dsc}\hat{T}_{dsc} + N_{bc}\hat{T}_{bc} + N_{dc}\hat{T}_{dc}}{\bar{T}} \tag{3.16}$$

$$K_m = \frac{N_{usm}\hat{T}_{usm} + N_{dsm}\hat{T}_{dsm} + N_{bm}\hat{T}_{bm} + N_{dm}\hat{T}_{dm}}{\bar{T}} \tag{3.17}$$

Table 3.4: Mean number of obstacle encounters and mean time penalties for eight obstacle classes

Obstacle Class	N_{obs}	$\hat{T}_{obs}$
Traversable Uphill Slopes	$\frac{D}{2\Delta}\text{erf}\left(\frac{\theta_{uphill,max}}{\sqrt{2\sigma^2}}\right)$	$\frac{k\Delta}{v\theta_{uphill,max}}\frac{2}{\text{erf}\left(\frac{\theta_{uphill,max}}{\sqrt{2\sigma^2}}\right)}\frac{\sigma}{\sqrt{2\pi}}\left(1-e^{-\frac{\theta_{uphill,max}^2}{2\sigma^2}}\right)$
Untraversable Uphill Slopes	$\frac{D}{2\Delta}\left[1-\text{erf}\left(\frac{\theta_{uphill,max}}{\sqrt{2\sigma^2}}\right)\right]$	$\frac{\Delta}{v}$
Traversable Downhill Slopes	$-\frac{D}{2\Delta}\text{erf}\left(\frac{\theta_{downhill,max}}{\sqrt{2\sigma^2}}\right)$	0
Untraversable Downhill Slopes	$\frac{D}{2\Delta}\left[1+\text{erf}\left(\frac{\theta_{downhill,max}}{\sqrt{2\sigma^2}}\right)\right]$	$\frac{\Delta}{v}$
Traversable Bumps or Ditches	$\frac{Le^{-sy^\star}(sy^\star+s(C+2b_w)+1)}{s}D$	$\frac{t^\star}{y^\star}\frac{sB+2-e^{-sy^\star}\left(s^2(C+2b_w)y^\star+s(C+2b_w)+s^2y^{\star 2}+2sy^\star+2\right)}{s\left(s(C+2b_w)+1-e^{-sy^\star}(s(C+2b_w)+sy^\star+1)\right)}$
Untraversable Bumps or Ditches	$\frac{Le^{-sy^\star}(sy^\star+s(C+2b_w)+1)}{s}D$	$\left(\frac{\pi-2}{2v}\right)\frac{e^{-sy^\star}\left(s^2(C+2b_w)y^\star+s(C+2b_w)+s^2y^{\star 2}+2sy^\star+2\right)}{se^{-sy^\star}(s(C+2b_w)+sy^\star+1)}$

3.2.3 Combined Equivalent Straight-Line Velocity (CESLV)

CESLV, a measure of the rate at which a team of robots can travel, is useful for predicting the performance of different robot team designs. Equation 3.18 shows that CESLV is the sum of the ESLVs of each robot in a team. CESLV provides measure of how fast a collection of waypoints (with known separation distances) could be visited by a team of robots. For example, if there are four pairs of waypoints that are 10 meters apart (40 total meters to travel), a team of four robots with a CESLV of 20 meters per minute could visit all the waypoints in 2 minutes. Notice that this example required making some assumptions, including that the mission would have to start with one robot on one of the two waypoints in each pair.

$$v_{cesl} = \sum_{team} v_{esl} \tag{3.18}$$

In the case of complex missions, CESLV provides an effective measure of mission performance without requiring the designer to determine complex mission plans for teams of different sizes. In other words, CESLV is a good measure of mission performance without requiring the mission plan to be known *a priori*. This is true for both static missions, where the mission plan is known and set prior to the mission, and dynamic missions, where the mission plan can change depending on measurements or observations.

Consider separating the performance of a static mission into three segments. During the first segment, the robot team is delivered to the planet's surface. The robots spread out from the landing site with each starting to perform its tasks at their respective locations. The first segment ends when all the robots have starting performing their work at full capacity. During the second segment, the robot team continues to perform their work at full capacity. During the third segment, at least one robot has finished its work, and the remainder of the team continues its work until each robot has finished. Figure 3.2 presents a mission timeline for a team of five robots with each segment highlighted.

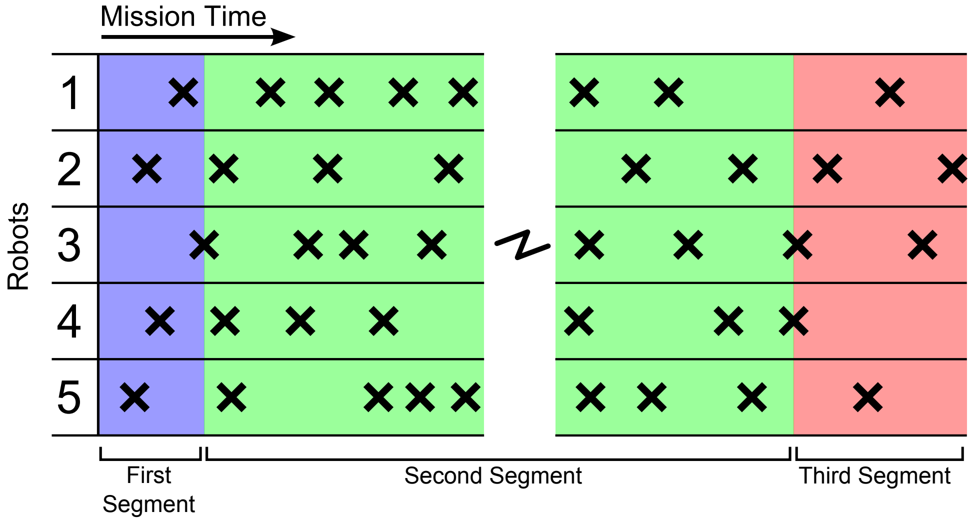


Figure 3.2: Mission timeline divided into three segments with "x" marks indicating when a task is performed. Spaces in-between "x" marks represent traveling time.

During the second segment, the team with the highest CESLV will perform the work the fastest. During the first and third segments, CESLV may not accurately represent the work rate of the team. However, for longer, complex missions the time spent on the first and third segments will be very small or negligible relative to the time spent on the second segment. Thus CESLV is representative of the speed at which the mission will be performed, regardless of the exact mission plan.

For dynamic missions, such as exploration missions, the team with the highest CESLV will be able to cover the most ground in a given amount of time. In general this would mean that the team with the highest CESLV would be able to perform the next chosen set of tasks fastest.

In order for designers to use CESLV as a performance metric for robot team designs, they need to account for several factors that CESLV does not account for. These factors originate from the effects that the choice of mission plan has on the performance of the team. However, after characterizing these factors and accounting for the time to complete tasks at each location, CESLV can provide an accurate estimate of mission completion time. This is a reason in favor of using this design framework as a way to explore the robot team tradespace, because designers can consider how effective different teams would be in light of the mission plan.

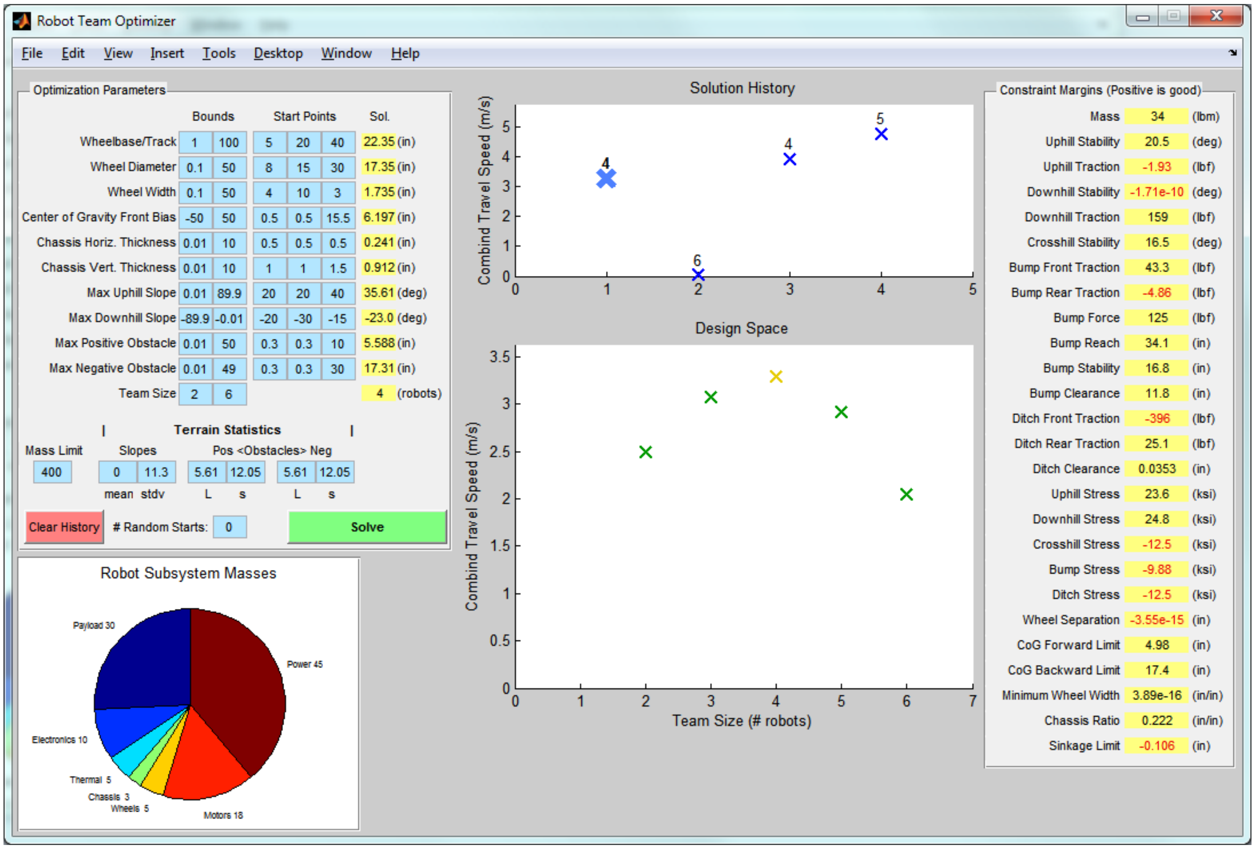
First, the designer should consider the maximum number of tasks that can be performed in parallel at one time. There is no advantage to using larger robot teams if robots will be idle. The mass of the extra robot could be more effectively allocated to improve the mobility of the rest of the team.

Second, the designer needs to consider the benefits of having robots cooperate on tasks. Some tasks could either require multiple robots or be completed faster with multiple robots. For example, teams that have a multiple of three robots may be advantageous in certain scenarios. The case study in Chapter 4 provides an example where having an even number of robots is advantageous.

Last, the designer must consider other requirements or constraints in the mission that are not included in CESLV. This includes other consequences of increasing or decreasing the number of robots in a team. For example, it could be significantly harder to maintain and track a larger team of robots. It could be difficult to pack a large team of robots for transportation through space. As a final example, it could be better to organize robots to collaborate on all tasks, so that a catastrophic failure in one robot does not prevent a task from being completed.

3.3 MATLAB Optimization Program

This design framework culminates to produce an optimization problem that returns the team with the highest CESLV. To illustrate this I developed a MATLAB program to solve the optimization. The program includes a graphical user interface, shown in Figure 3.3, to aid in tradespace exploration. The program uses MATLAB's MultiStart solver with a sequential quadratic programming (SQP) algorithm to solve the optimization.



Optimization Parameters

	Bounds		Start Points			Sol.	
Wheelbase/Track	1	100	5	20	40	22.35	(in)
Wheel Diameter	0.1	50	8	15	30	17.35	(in)
Wheel Width	0.1	50	4	10	3	1.735	(in)
Center of Gravity Front Bias	-50	50	0.5	0.5	15.5	6.197	(in)
Chassis Horiz. Thickness	0.01	10	0.5	0.5	0.5	0.241	(in)
Chassis Vert. Thickness	0.01	10	1	1	1.5	0.912	(in)
Max Uphill Slope	0.01	89.9	20	20	40	35.61	(deg)
Max Downhill Slope	-89.9	-0.01	-20	-30	-15	-23.0	(deg)
Max Positive Obstacle	0.01	50	0.3	0.3	10	5.588	(in)
Max Negative Obstacle	0.01	49	0.3	0.3	30	17.31	(in)
Team Size	2	6				4	(robots)

Terrain Statistics

Mass Limit	Slopes		Pos <Obstacles>		Neg	
400	0	11.3	5.61	12.05	5.61	12.05
	mean	stdv	L	s	L	s

Random Starts: 0

Constraint Margins (Positive is good)

Mass	34	(lbm)
Uphill Stability	20.5	(deg)
Uphill Traction	-1.93	(lbf)
Downhill Stability	-1.71e-10	(deg)
Downhill Traction	159	(lbf)
Crosshill Stability	16.5	(deg)
Bump Front Traction	43.3	(lbf)
Bump Rear Traction	-4.86	(lbf)
Bump Force	125	(lbf)
Bump Reach	34.1	(in)
Bump Stability	16.8	(in)
Bump Clearance	11.8	(in)
Ditch Front Traction	-396	(lbf)
Ditch Rear Traction	25.1	(lbf)
Ditch Clearance	0.0353	(in)
Uphill Stress	23.6	(ksi)
Downhill Stress	24.8	(ksi)
Crosshill Stress	-12.5	(ksi)
Bump Stress	-9.88	(ksi)
Ditch Stress	-12.5	(ksi)
Wheel Separation	-3.55e-15	(in)
CoG Forward Limit	4.98	(in)
CoG Backward Limit	17.4	(in)
Minimum Wheel Width	3.89e-16	(in/in)
Chassis Ratio	0.222	(in/in)
Sinkage Limit	-0.106	(in)

Figure 3.3: Graphical user interface for comparing optimizations with different parameters

SQP is a powerful routine that is designed to solve constrained non-linear optimization programs. SQP follows an iterative solving process where it approximates the optimization problem as a quadratic optimization problem. SQP begins at a given starting point, solves the approximate quadratic problem at that point, and repeats the procedure at the new solution point.

It is very likely that the resulting optimization problem from the models is non-convex, meaning that there are several local optima. In that case, SQP may return a local optima instead of the global optimum. To compensate for this, global optimization solvers, like MATLAB's MultiStart solver, run multiple SQP optimizations at different starting points. Rather than using the solver with a set of random starting points, a designer could use heuristic knowledge to select advantageous starting points.

Although using SQP on multiple starting points is effective for solving for the optimal robot team design, it should be noted that SQP cannot directly solve for the optimal number of robots in a team. This is because the number of robots in a team is a discrete variable: it can only have positive integer values. There are two ways to circumvent this problem.

First, multiple global SQP optimizations can be performed for different given team sizes. In most cases, the relationship between team size and CESLV is convex, so the optimal team size can be found through a guess and check procedure of running optimizations with different size teams.

Second, a global SQP optimization can be initially run to optimize ESLV divided by mass for one robot. The result is the most mass-efficient robot design for the given terrain conditions. For a given mass budget, there will generally be a non-integer number of those robots that fill the mass budget. Because of the convex relationship between CESLV and team size, the optimal team size will be one of the two nearest integers to that number. Thus, only two optimizations need to be run to determine the best robot team design.

3.3.1 Program Structure

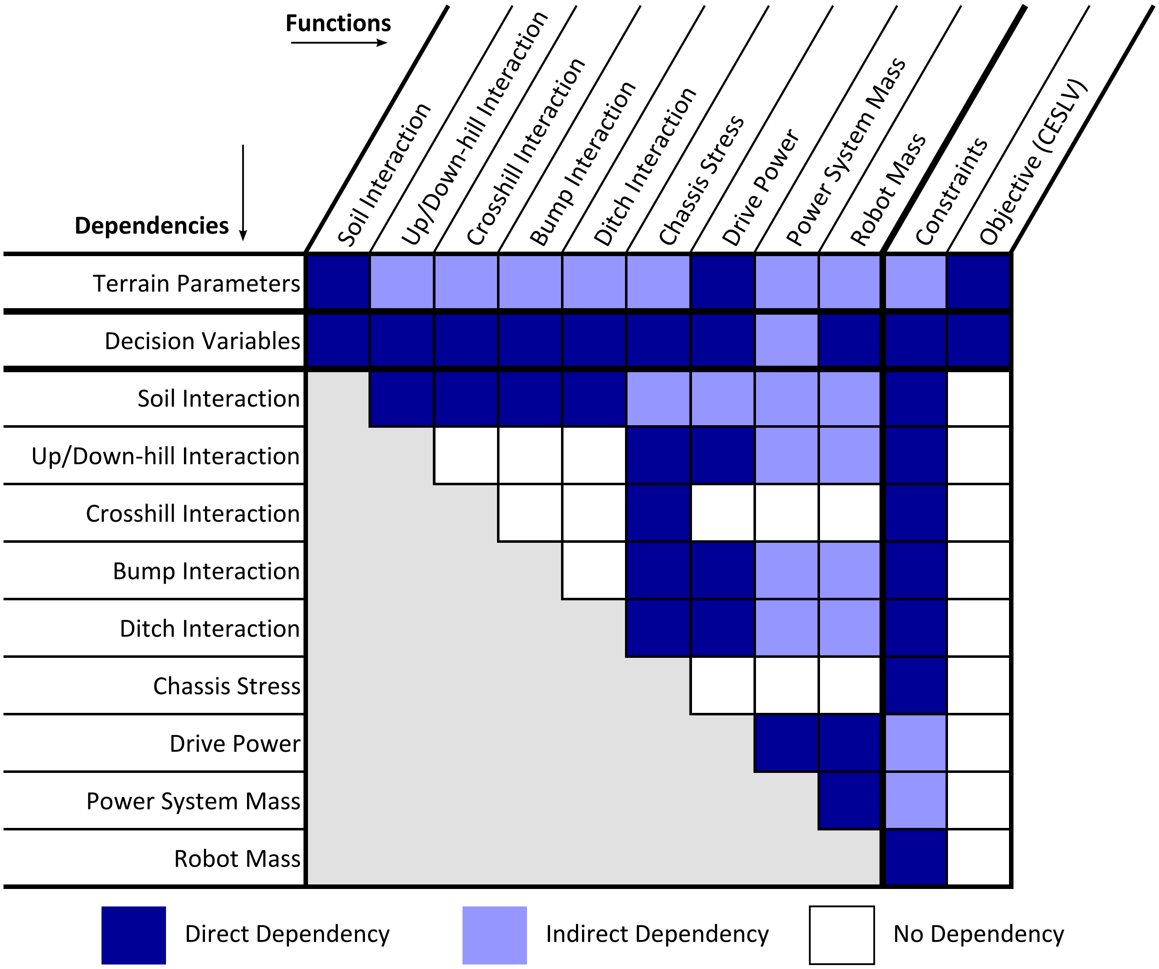
The MATLAB program relies on several functions that contain different aspects of the models presented in Chapter 2. Functions are an effective way to implement any relationships that are reused in multiple parts of the models. Table 3.5 presents the functions in the MATLAB program along with their inputs and outputs. Listed in the inputs, "Terrain Parameters" refers to the obstacle sizes and distributions and the set of values to describe the soil conditions. See Table 4.1 and Table 4.2 for a list of those parameters.

Many of the MATLAB functions implement one or more of the other functions. Figure 3.4 shows the dependencies between functions. Functions are listed at the top, with required functions or required parameters highlighted in the columns below. Dark blue fill represents a direct dependency, in which the top function includes calls to the other functions or uses the parameters. Light blue fill represents an indirect dependency, in which one of the dependent functions uses the function or parameters.

During an optimization, MATLAB's SQP routine makes many calls to the Objective function and the Constraints function. The Objective and Constraints functions will call any necessary

Table 3.5: Functions in the MATLAB Optimization Program

Function Name	Description	Inputs	Outputs
Soil Interaction	Determines the tractive and resistive forces generated at the wheel-soil interface	terrain parameters, wheel dimensions, wheel loading	sinkage, net motion resistance, net tractive effort
Uphill/Downhill Slope Interaction	Determines the stability margin of the robot and the robot's drawbar pull for a given slope	slope angle, robot wheelbase, CoG location	slope motion resistance, robot stability margin
Crosshill Interaction	Determines the stability margin of the robot and the robot's drawbar pull for a given crosshill slope	robot track, wheel width, CoG location	robot stability margin, drawbar pull
Bump Interaction	Determines the forces and the three margins (see Figure 2.11) for a robot traversing a bump	bump height, robot dimensions, terrain parameters	robot stability margin, robot force margins, geometrical margins
Ditch Interaction	Determines the forces and the two factors (see Figure 2.12) for a robot traversing a ditch	ditch width, robot dimensions, terrain parameters	robot force margins, geometrical margins
Chassis Stress	Determines the stresses in the robot's chassis when traversing a given obstacle	robot dimensions, robot mass, chassis material strength, obstacle type and magnitude	yield stress margin
Drive Power	Determines the average power the robot uses for driving, based on the distribution of obstacles	terrain parameters, maximum obstacle sizes, robot dimensions	average required drive power
Power System Mass	Determines the mass of the solar panels, batteries, and power management electronics	average required drive power	power system mass
Robot Mass	Determines the total mass of the robot	robot dimensions, power system mass, drive power	total robot mass
Optimization Constraints	Checks if all the constraints are met	robot dimensions, sinkage, obstacle margins, chassis stress margins	constraint margins
Optimization Objective	Determines the CESLV of the robot team	robot dimensions, maximum obstacle magnitudes, terrain data	ESLV, CESLV



Dependencies ↓ / Functions →	Soil Interaction	Up/Down-hill Interaction	Crosshill Interaction	Bump Interaction	Ditch Interaction	Chassis Stress	Drive Power	Power System Mass	Robot Mass	Constraints	Objective (CESLV)
Terrain Parameters	■	▨	▨	▨	▨	▨	■	▨	▨	▨	■
Decision Variables	■	■	■	■	■	■	■	▨	■	■	■
Soil Interaction		■	■	■	■	▨	▨	▨	▨	■	□
Up/Down-hill Interaction			□	□	□	■	■	▨	▨	■	□
Crosshill Interaction				□	□	■	□	□	□	■	□
Bump Interaction					□	■	■	▨	▨	■	□
Ditch Interaction						■	■	▨	▨	■	□
Chassis Stress							□	□	□	■	□
Drive Power								■	■	▨	□
Power System Mass									■	▨	□
Robot Mass										■	□

■ Direct Dependency ▨ Indirect Dependency □ No Dependency

Figure 3.4: Dependencies between MATLAB functions.

dependency functions. Those dependency functions will call other dependency functions, and so on. The decision variables, along with all the constant parameters that the models use, are passed through each dependency function.

3.3.2 Graphical User Interface

Shown in Figure 3.3, the graphical user interface (UI) for the program allows for different size robot teams to be easily compared. This is especially useful for choosing the size of a team when other factors (not incorporated into the optimization) have a significant impact on the design choice.

The GUI displays the terrain data, decision variables, constraints, and robot subsystem masses. A history of CESLVs is shown for previously run optimizations. Previous optimization runs and alternative designs shown on the graphs can be selected with the mouse to view their decision variables, constraint margins, and subsystem masses. A designer could quickly compare and contrast designs to decide which are practical based on external factors.

Chapter 4

Case Study - Mars Exploration

This chapter presents the process of using the design framework developed in Chapters 2 and 3 to design a team of robots for an example mission on Mars. First, the mission scenario is presented. Second, the design framework is used within the context of the mission to determine the best design for the robots. Finally, a sensitivity analysis is performed to analyze the impact that different mission conditions have on the optimal design.

4.1 Scenario

For this mission, a team of robots is to explore a region near the NASA Viking Lander 1 site by traveling along eight paths originating from a single starting point, as shown in Figure 4.1. As they travel, the robots will take two core samples at 30 sites along each path. Collecting one sample requires 10 minutes. Robots may depart from the path to avoid obstacles, but must return to the sites on the original path for sampling. Sites are evenly spaced 400 meters apart. The first site on each path is 400 meters from the center intersection of all the paths. The mission is complete once all the core samples have been collected.

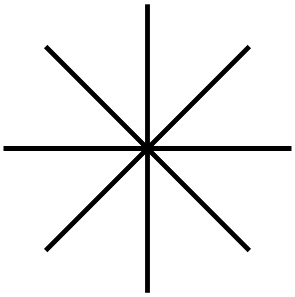


Figure 4.1: For the case study mission, measurements are to be taken along the paths shown above

Tables 4.1 and 4.2 list the terrain data for the region where the mission will take place. The soil data is from a combination of sources, including measurements from the Viking Lander 1 site. Parameters that are not available are from soils with properties similar to those known for the

Viking Lander 1 site soil.

Table 4.1: Soil parameters for the Mars case study

Parameter	Symbol	Value	Source
Cohesion	c	0.3 psi	Chosen based on a several measurements near the Viking Lander 1 site [48]
Angle of internal friction	ϕ	30°	
Cohesive modulus	k_c	$2.3\ \frac{\text{lbf}}{\text{in}^{n+1}}$	Values for sandy loam, which has similar cohesion and angle of internal friction to the Martian values [30]
Frictional modulus	k_ϕ	$16.8\ \frac{\text{lbf}}{\text{in}^{n+2}}$	
Exponent of sinkage	n	0.7	
Slip coefficient	K	0.4 in	Value for firm sandy soil [30]
Unit weight	γ	$0.0208\ \frac{\text{lbf}}{\text{in}^3}$	Chosen based on a several measurements near the Viking Lander 1 site [48, 49]

The slope distribution is determined from Mars elevation data. The elevation values are measured along evenly spaced increments of one meter in each direction. The slope distribution is generated by calculating the elevation change per resolution (one meter) in the longitudinal and latitudinal directions for every pair of neighboring elevation measurements (see Figure 4.2). Figure 4.3 shows that a Gaussian distribution represents the distribution of slopes with reasonable accuracy.

Table 4.2: Terrain obstacle parameters for the Mars case study

Obstacle	Parameter	Symbol	Value	Source
Slope	Mean	μ	0°	Mars elevation data near the Viking Lander 1 site [50]
	Standard Deviation	σ	24.98°	
Bump	Exponential Parameter	s	$12.05\,\text{m}^{-1}$	Rock size and frequency measurements from the Viking Lander 1 site [46]
	Density Parameter	L	$5.61\,\text{m}^{-2}$	
Ditch	Exponential Parameter	s	$12.05\,\text{m}^{-1}$	Assumed to be the same as the bump parameters
	Density Parameter	L	$5.61\,\text{m}^{-2}$	

The size-frequency distribution for rocks at the Viking Lander 1 site is used as the parameters for bump obstacles. Assuming that the distribution of positive and negative obstacles are similar allows the same parameters to be used for ditch obstacles.

The mass budget for this mission is 1000 lbm. Appendix E provides other values used to complete the models for the optimization. One notable assumption is that the payload mass of the robot is 30 pounds, which includes the mass of the instruments required to collect core samples.

4.2 Results and Discussion

I formulate an optimization problem (see Chapter 3) from the models developed in Chapter 2 and the data in Section 4.1. Figure 4.4 presents the results from the optimization. For this scenario,

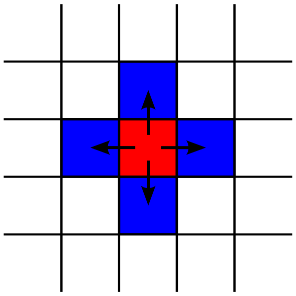


Figure 4.2: Slopes are calculated by comparing each data point to its four neighboring points

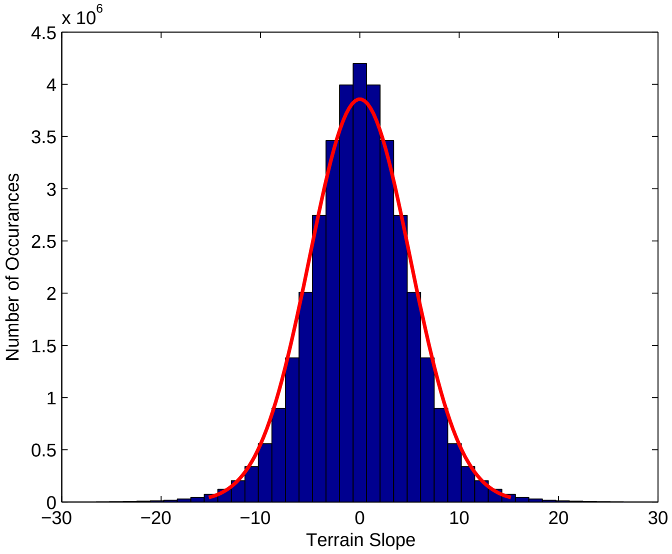


Figure 4.3: Histogram of slope angles from Viking Lander 1 site on Mars fit to Gaussian curve

multi-robot teams provide superior performance to a single robot, as measured by their CESLVs. A team of 11 robots provides the best CESLV of 7.95 meters per second.

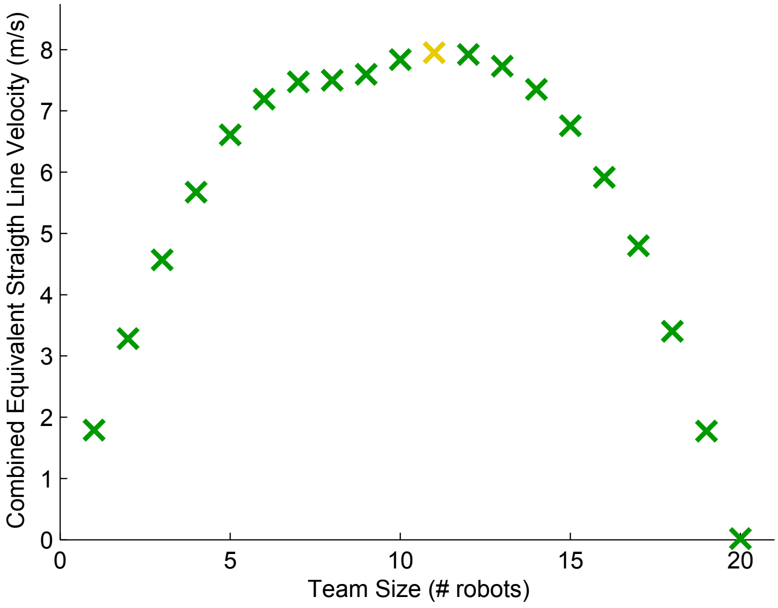


Figure 4.4: Design tradespace for the case study

The tradespace generated via this design framework presents a phenomenon that may otherwise go unnoticed to a designer. Figure 4.4 shows that there are two distinct trends for the optimal design of individual robots, depending on the number of robots in the team. If a designer had determined the optimal design proportions for five robots, for example, scaling the design for a team with more than 10 robots would give a suboptimal team.

The shift in trends is evident from the change in the curvature of the data points. Figure 4.5 shows that the shift in trends occurs between teams of 8 and 9 robots. The shift is characterized by the reallocation of mass from the chassis and motors to the wheels.

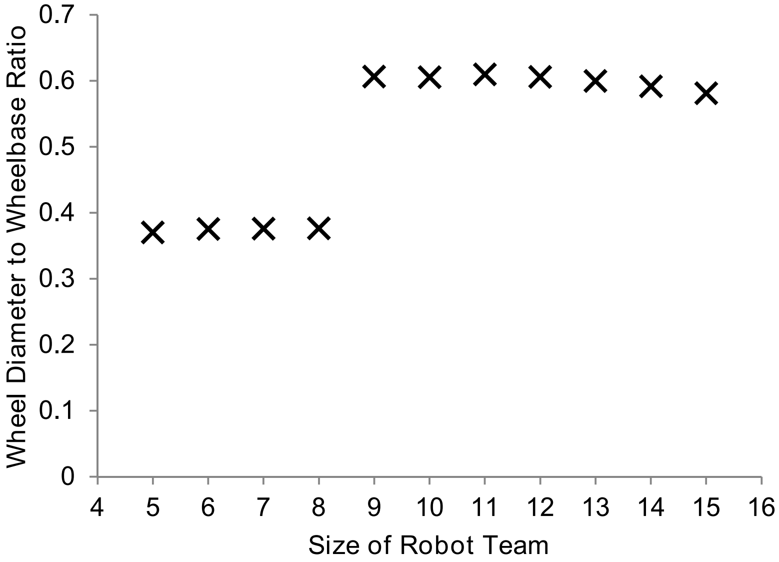


Figure 4.5: Tradespace trends in the ratio of wheel size to robot size

The reason for this shift is the mass-cost associated with the ability of the robot to overcome bump obstacles. Figure 4.6 shows that, for the first trend, the ability of the robot to traverse bump obstacles decreases at a faster rate than the abilities to traverse other types of obstacles. The shift indicates that a large portion of the effort (in terms of mass) spent on the ability to traverse bump obstacles would be more beneficial elsewhere when incrementing from eight to nine robots. Specifically, that mass is more beneficial when spent on the abilities of robots to traverse ditch and slope obstacles.

In the context of the mission, there are three important factors to consider when trying to determine the best design for a robot team.

First, for this mission, there are eight distinct paths to travel and two core samples are required at each site of interest. A team of eight robots could assign one robot to each path. It is also possible that two robots could split the sampling sites along the same path. This would require 16 robots to work each path simultaneously. Because of this, a team of 8 or 16 robots has the advantage that it does not require robots to travel from one path to another. Team sizes that have large least common multiples with 8 (the number of paths) would require more crossover between paths, which may increase the longest path that any robot is required to travel.

Second, when two robots are traveling along the same path, the full degree of the teams's CESLV is not utilized. With regard to the mission, they are performing at least some amount of redundant work by covering the same ground, even though both may be collecting core samples at each site. Thus, even though teams with more than 8 robots have high CESLVs, a designer needs to

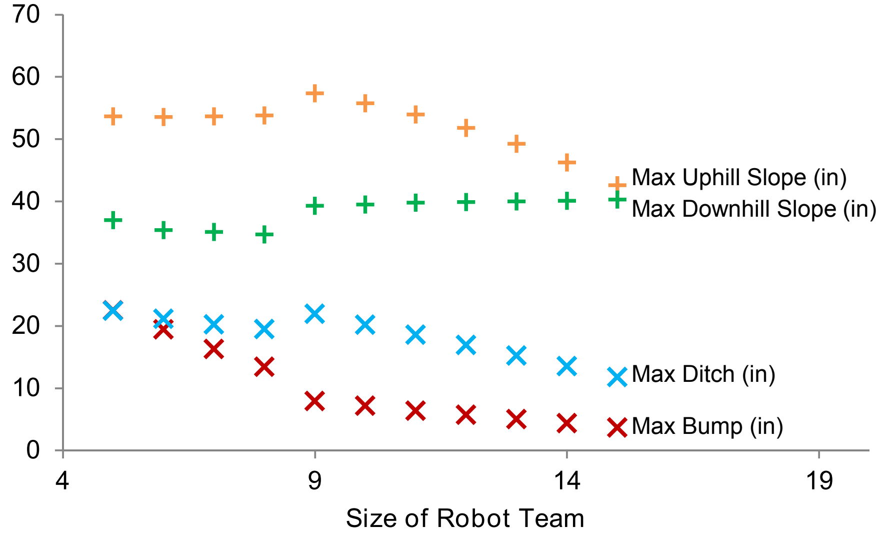


Figure 4.6: Tradespace trends in maximum traversable obstacles magnitudes

understand that mission plans for those teams may not take full advantage of the team's CESLV. In the end, it is possible that teams with higher CESLVs may take longer to perform the mission because of this. This phenomenon is unique to highly structured missions, like this one, in which a relatively small number of tasks have a very limited number of practical ways that they could be performed. Fortunately for missions like this, mission plans for teams of different sizes are much easier to devise than in the case of larger missions, which do not experience this phenomenon in the first place.

Third, the time required to collect each core sample could impact the relative performance of the robot teams in the tradespace. Short core sampling times favor smaller teams while long core sampling times favor larger teams. This can be understood by considering two extreme cases. For extremely short sample times, the mission performance is primarily influenced by the time required for robots to reach the furthest sample site. For extremely long sample times, the mission performance is primarily influenced by the time to collect samples. In the latter case, larger teams would be able to work on more samples simultaneously.

Although it is possible to estimate mission completion times by determining ideal mission plans for each possible team design, considering the above factors, a team with 12 robots will likely be best. The 12-robot team has the second highest CESLV of 7.923 m/s, which is 0.3% smaller than the optimum CESLV for the team of 11 robots. Unlike the 11-robot team, the 12-robot team can easily be divided to work on the eight paths, as evident by their relatively low least common multiple of 24. The 11-robot team would require additional crossover between paths, as evident by their relatively high least common multiple of 88. Other teams with sizes close to 12 also have higher least common multiples. Table 4.3 lists the design parameters and the subsystem masses for the twelve-robot team.

Table 4.3: Design parameters and subsystem masses for the eight-robot team

Parameter	Value
Wheelbase/Track	27.96 in
Wheel Diameter	16.94 in
Wheel Width	1.694 in
CoG Bias	4.378 in
Chassis Horizontal Thickness	0.325 in
Chassis Vertical Thickness	0.482 in
Maximum Uphill Slope	51.81°
Maximum Downhill Slope	39.90°
Maximum Bump Height	5.678 in
Maximum Ditch Width	16.89 in

Subsystem	Mass (lbs)
Payload	30
Electronics	10
Thermal Management	5
Chassis	3
Wheels	6
Motors	5
Power Supply	12

4.3 Robustness to Changing Mission Conditions

It is important for a designer to understand the influence that different mission conditions have on the optimal robot team design. Errors in mission condition data due to misrepresentative measurements, changes in mission location, or other factors can significantly impact the optimal robot-team designs. This section will examine the sensitivity of the tradespace of robot team designs to changes in the mission conditions.

This sensitivity analysis considers the effect of changes in mass budget, slope standard deviation, and both bump distribution parameters on the robot team design tradespace. Figure 4.7 compares the tradespaces for increases and decreases in each mission parameter by 10 percent and 25 percent.

Changes in the mass budget of the robot team shift the size of the optimal team. At the extremes, a 25% decrease in the mass budget results in an optimal team of 8 robots, compared to the original optimum of 11 robots. A 25% increase results in an optimal team of 14 robots. This shift is due to the optimization driving the robot design to be as close as possible to the most mass-efficient robot design.

Despite this shift in the size of the optimal team, designs for smaller robot teams are only slightly affected by changes in the mass budget. First, the CESLV of small teams (less than five robots) does not change significantly for changes to the mass budget. Second, the relative difference between the CESLV of smaller teams (less than 10 to 15 robots, depending on the change in mass budget) and the CESLV of the optimal team is not significantly affected by changes to the mass budget. The relevance of this is that post-design changes to the mass budget would not require as much redesigning of the robot team and mission plan if a smaller team is initially selected.

Changes in the standard deviation of the slope or in the bump distribution parameters do not significantly shift the size of the optimal robot team. Instead, those changes primarily scale the tradespace vertically. This means that the optimal robot team design is robust to errors in terrain parameters, which is desirable to a designer.

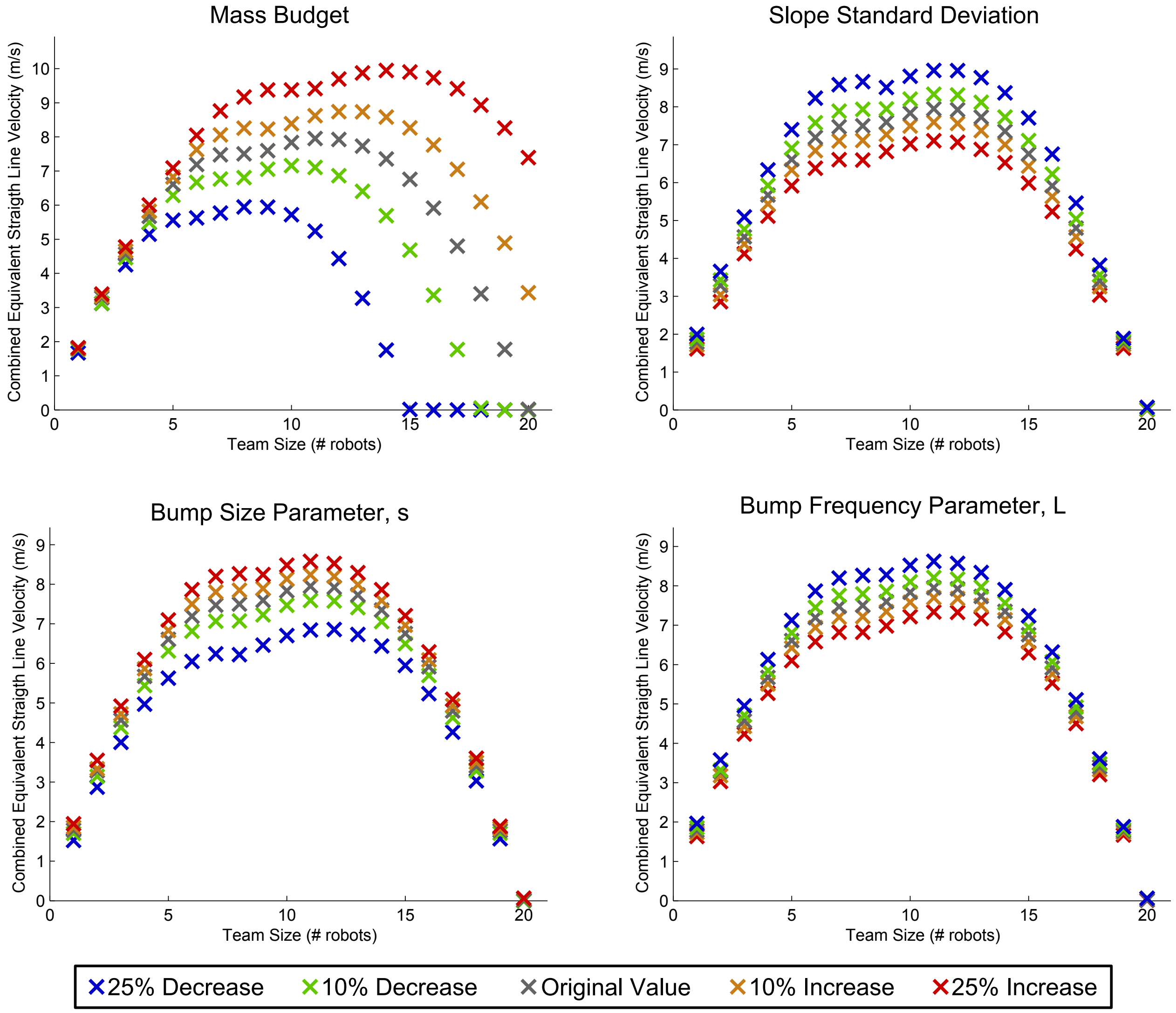


Figure 4.7: Mars mission tradespaces for 10% and 25% decreases and increases in different mission parameters

Chapter 5

Summary and Conclusions

This thesis presents the development of a design framework for determining the design of a team of robots that has a limited mass budget and that will operate on uncertain terrain conditions. Chapter 2 presents a model of the terrain, a model of the robot, and a model of the robot-terrain interactions as part of the framework. Chapter 3 develops an optimization problem from the models that can be solved to determine the optimal design for a team of robots. The optimization problem is based on a new measure of mission performance for teams of robots. Chapter 4 presents a case study on the design of a team of robots for a mission to Mars. The current chapter summarizes the work in this thesis, lists the contributions of this thesis, and presents suggestions for future work.

5.1 Conclusions

Although the frequency of planetary exploration missions with mobile robots is increasing, the cost of these missions remains very high. A popular idea to reduce the cost of these missions is to distribute the mass allowance for a mission over a team of robots, instead of using a single robot. This thesis develops a design framework that examines the tradeoffs between the size of the robot team and the mobility of the individual robots for a robot team operating on uncertain terrain conditions. The framework allows a designer to explore the tradespace of robot team sizes and determines the robot team that provides the best performance based on CESLV, a novel performance metric.

The design framework is built on models of the robots, models of the terrain hazards, and models for the interaction of the robot with the terrain, presented in Chapter 2. The robot is modeled using a simplified geometry of a four-wheeled robot. The terrain hazards are categorized into three types: slope obstacles, positive obstacles, and negative obstacles. The magnitude and frequency of the obstacles are modeled using statistical distributions. The model for terrain-obstacle interactions uses Bekker Theory to predict the terrain-wheel forces. To be able to traverse a given obstacle, the robot has to meet three requirements. First, the robot must be able to generate enough force at the wheel-soil interface. Second, the robot must not roll or pitch over. Third, the body of the robot must not collide with the obstacle during the traverse. A designer could modify or supplement these models as desired to achieve greater complexity or accuracy.

The models can be formulated into an optimization problem, and then solved to determine the optimal robot team design. To determine the optimal designs, the optimization uses a novel measure of mission performance for a team of mobile robots, known as the combined equivalent straight-line velocity (CESLV). CESLV is the sum of the equivalent straight-line velocities ESLV of each robot in the team. ESLV is a measure of the average rate that a robot can travel between two locations based on the magnitude and frequency of obstacles in the terrain model. CESLV is an effective measure of mission performance that is resistant to the impacts that mission planning can have on the actual performance of the mission. CESLV is applicable to static missions with a full plan known pre-launch, or dynamic missions where the plan changes depending on observations during the mission. Additionally, CESLV is an effective aid for selecting the design of a robot team in the case of a highly structured mission.

Chapter 4 presents a case study for the design of a team of robots to perform a structured mission on Mars. Terrain data for the case study is taken primarily from measurements at the Viking Lander 1 site on Mars. The case study shows how CESLV can aid in selecting a robot team design in the case of a mission with specific requirements. A graphical user interface (GUI), developed in MATLAB, allows a designer to explore the tradespace of possible robot teams. This allows the designer to consider that teams with suboptimal CESLVs might be best in light of influences that are not captured by the models. The design framework indicates that a team of robots provides superior performance to a single robot for the case study mission. It also shows that the optimal size of the team is robust to changes in the terrain conditions, but sensitive to changes in the mass budget. The ability of the design framework to provide valuable insight in both static and dynamic mission plans, as well as in highly structured mission plans, makes it a valuable tool to mission designers.

5.2 Suggestions for Future Work

Designing and planning for planetary exploration missions with mobile robots is complex. The ultimate goal of this work would be to have an objective procedure for determining both the design of a team of robots and their mission plan for a given mission scenario. There are several areas for additional work that could improve or add to the design framework presented in this thesis.

CESLV is a representative measure of mission performance that is independent of the effects of mission planning in some cases, and resistant to the impacts of mission planning in other cases. A method to represent a mission in such a way that a computer algorithm could quickly determine an optimal mission would improve this design framework. For a designer, this addition would remove some subjectivity from the design process.

This thesis considers designing homogeneous teams of robots. However, heterogeneous teams may be advantageous for many missions. For heterogeneous teams there is a stronger dependence between the design of the team and the mission plan. The best robot team for a mission would need to have both an optimal design and an optimal mission plan. Future research could develop a way to consider the tasks that need to be completed in the mission and the functions that different robot components can perform. A design framework for heterogeneous teams would have to include a method for determining the mission plan for a given robot team.

For mobile robots operating in remote environments, component failure or immobilization from terrain conditions could delay the mission or prevent it from being completed. For large missions that use dozens of robots, there is a higher probability that one or multiple robots fail. Designing the team of robots or the mission plan while accounting for the probability of failure could affect the size and design of the best team for the mission.

Finally, this thesis uses simple models to focus on demonstrating the overall framework. More complex models could improve the accuracy of the design tradespace that is generated by the optimization. Improved terramechanics or robot-obstacle interaction models would benefit this design framework. Robot subsystem models could be improved, and additional robot locomotion systems could be modeled.

Bibliography

[1] NASA. Mars Pathfinder Landing. Technical report, NASA, 1997.

[2] NASA. Mars Exploration Rover Landings. Technical report, NASA, 2004.

[3] NASA. Mars Science Laboratory Landing. Technical report, NASA, 2012.

[4] ISECP. The Global Exploration Roadmap. Technical report, International Space Exploration Coordination Group, 2013.

[5] Andrew Moseman. 4 Things to Know About NASA's Next Mars Rover, 2012.

[6] Stephen Clark. Europe's ExoMars May End Russia's Bad Luck on Mars, 2013.

[7] Patricia M. Beauchamp, James a. Cutts, Marco Quadrelli, Lincoln Wood, Joseph E. Riedel, Michael C. McHenry, MiMi Aung, Richard Volpe, and Laureano Cangahuala. Guidance Navigation and Control Technology Assessment for Future Planetary Science Missions. *AIAA SPACE 2013 Conference and Exposition*, pages 1–26, September 2013.

[8] JA Lamamy and D Miller. *Enhancing the science return of Mars missions via sample preparation, robotic surface exploration and in orbit fuel production.* PhD thesis, Massachusetts Institute of Technology, 2004.

[9] JA Lamamy and DW Miller. Designing the Next Generation of Rovers through a Mid-rover Analysis. *ASTRA*, 2006.

[10] JA Lamamy and DW Miller. *Methods and tools for the formulation, evaluation and optimization of rover mission concepts.* PhD thesis, Massachusetts Institute of Technology, 2007.

[11] Farah Alibay, Vishnu R Desaraju, Jessica E Duda, and Jeffrey A Hoffman. Acta Astronautica Fractionated robotic architectures for planetary surface mobility systems. *Acta Astronautica*, 95:15–29, 2014.

[12] Aurora. Multi-Robot Planetary Exploration Architectures - Phase II STTR. Technical report, 2012.

[13] EG Johnsen. Man, Teleoperators, and Robots: An Optimum Team for Space Exploration. *Journal of Spacecraft and Rockets*, 9(7):554–556, 1972.

[14] Amy Bilton, Yoshiyuki Ishijima, Matthew Lichter, and Steven Dubowsky. Sensor architecture for the robotic control of large flexible space structures. *Journal of Field Robotics*, 24(4):297–310, April 2007.

[15] Ashley Stroupe, Terry Huntsberger, Avi Okon, and Hrand Aghazarian. Precision manipulation with cooperative robots. In *Multi-Robot Systems. From Swarms to Intelligent Automata Volume III*, volume III, pages 235–248. Springer, 2005.

[16] Ashley Stroupe, Avi Okon, Matthew Robinson, Terry Huntsberger, Hrand Aghazarian, and Eric Baumgartner. Sustainable cooperative robotic technologies for human and robotic outpost infrastructure construction and maintenance. *Autonomous Robots*, 20(2):113–123, April 2006.

[17] S Sand, S Zhang, and M Muhlegg. Swarm exploration and navigation on Mars. In *Localization and GNSS (ICL-GNSS), 2013 International Conference on*, pages 25–27, 2013.

[18] David Michel and Kenneth McIsaac. New path planning scheme for complete coverage of mapped areas by single and multiple robots. *2012 IEEE International Conference on Mechatronics and Automation*, pages 1233–1240, August 2012.

[19] T Huntsberger, P Pirjanian, a Trebi-Ollennu, H Das Nayar, H Aghazarian, a J Ganino, M Garrett, S S Joshi, and P S Schenker. CAMPOUT: a control architecture for tightly coupled coordination of multirobot systems for planetary surface exploration. *IEEE Transactions on Systems Man and Cybernetics Part A Systems and Humans*, 33(5):550–559, 2003.

[20] Prithviraj Dasgupta, Taylor Whipple, and Ke Cheng. Effects of Multi-robot Team Formations on Distributed Area Coverage. *International Journal of Swarm ...*, 2011.

[21] Giuliano Punzo, Gordon Dobie, and DJ Bennet. Low-cost, multi-agent systems for planetary surface exploration. *63rd International Astronautical Congress*, 2012.

[22] SB Stancliff. Mission Reliability Estimation for Repairable Robot Teams. *International Journal of Advanced Robotic Systems*, 3(2):155–164, 2006.

[23] Adamantios Mettas. Reliability allocation and optimization for complex systems. In *Reliability and Maintainability Symposium*, pages 216–221. Ieee, 2000.

[24] C Bererton and P Khosla. An analysis of cooperative repair capabilities in a team of robots. *Proceedings 2002 IEEE International Conference on Robotics and Automation Cat No02CH37292*, 1(May):476–482, 2002.

[25] David Asikin and JM Dolan. A mission taxonomy-based approach to planetary rover cost-reliability tradeoffs. In *Proceedings of the 9th Workshop on Performance Metrics for Intelligent Systems*, pages 49–56, 2009.

[26] M. G. Bekker. *Theory of Land Locomotion.* The University of Michigan Press, Ann Arbor, MI, 1956.

[27] M. G. Bekker. *Introduction to Terrain-Vehicle Systems.* The University of Michigan Press, Ann Arbor, MI, 1969.

[28] Karl Terzaghi. *Theoretical Soil Mechanics.* John Wiley & Sons, Inc., New York, NY, 1943.

[29] Karl Terzaghi, Ralph Peck, and Gholamreza Mesri. *Soil Mechanics in Engineering Practice.* John Wiley & Sons, Inc., New York, NY, 3 edition, 1996.

[30] J. Y. Wong. *Theory of Ground Vehicles.* John Wiley & Sons, Inc., Hoboken, NJ, 4 edition, 2008.

[31] Liang Ding, Haibo Gao, Zongquan Deng, Keiji Nagatani, and Kazuya Yoshida. Experimental study and analysis on driving wheels' performance for planetary exploration rovers moving in deformable soil. *Journal of Terramechanics*, 48(1):27–45, February 2011.

[32] Masataku Sutoh, Junya Yusa, and Tsuyoshi Ito. Traveling performance evaluation of planetary rovers on loose soil. *Journal of Field Robotics*, 29(2010):648–662, 2012.

[33] Krzysztof Skonieczny, Scott J. Moreland, and David S. Wettergreen. A grouser spacing equation for determining appropriate geometry of planetary rover wheels. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5065–5070, October 2012.

[34] J Liu, H Gao, and Z Deng. Effect of straight grousers parameters on motion performance of small rigid wheel on loose sand. *Information Technology Journal*, 2008.

[35] Dimitrios Apostolopoulos. *Analytic Configuration of Wheeled Robotic Locomotion*. PhD thesis, Carnegie Mellon University, 2001.

[36] Matthew D. Berkemeier, Eric Poulson, and Travis Groethe. Elementary mechanical analysis of obstacle crossing for wheeled vehicles. *2008 IEEE International Conference on Robotics and Automation*, pages 2319–2324, May 2008.

[37] BSC Ranjan and Ujjwal Pal. Exploring Serially Connected Multi-Tracked All-Terrain Vehicles for Improved Obstacle Climbing Performance. *14 th National Conference on Machines and Mechanisms (NaCoMM09)*, pages 279–286, 2009.

[38] Amir H. Rajabi, Amir H. Soltanzadeh, Arash Alizadeh, and Golnaz Eftekhari. Prediction of obstacle climbing capability for tracked vehicles. *2011 IEEE International Symposium on Safety, Security, and Rescue Robotics*, (c):128–133, November 2011.

[39] Peng Zhang, Zongquan Deng, Ming Hu, and Haibo Gao. Mobility performance analysis of lunar rover based on terramechanics. In *International Conference on Advanced Intelligent Mechatronics*, number 2006, pages 120–125, 2008.

[40] Nildeep Patel, Alex Ellery, Elie Allouis, Martin Sweeting, and Lutz Richter. Rover mobility performance evaluation tool (RMPET): a systematic tool for rover chassis evaluation via application of Bekker theory. In *Proceedings of the 8th ESA Workshop on Advanced Space Technologies for Robotics and Automation*, 2004.

[41] Brian Wilcox, A Nasif, and Rick Welch. Implications of martian rock distributions on rover scaling. 1997.

[42] S Michaud, L Richter, and T Thuer. Rover chassis evaluation and design optimization using the RCET. In *The 9th ESA Workshop on Advanced Space Technologies for Robotics and Automation (ASTRA'06)*, number 1, 2006.

[43] Thomas Thueer and Roland Siegwart. Mobility evaluation of wheeled all-terrain robots. *Robotics and Autonomous Systems*, 58(5):508–519, May 2010.

[44] JJ Biesiadecki and MW Maimone. The mars exploration rover surface mobility flight software driving ambition. *IEEE Aerospace Conference*, 2006.

[45] Frederick Jindra. Obstacle performance of articulated wheeled vehicles. *Journal of Terramechanics*, 3(2), 1966.

[46] M. Golombek and D. Rapp. Size-frequency distributions of rocks on Mars and Earth analog sites: Implications for future landed missions. *Journal of Geophysical Research*, 102(E2):4117, February 1997.

[47] M. G. Bekker. *Off-the-Road Locomotion.* The University of Michigan Press, Ann Arbor, MI, 1960.

[48] HJ Moore, GD Clow, and RE Hutton. A summary of Viking sample-trench analyses for angles of internal friction and cohesions. *Journal of Geophysical Research*, 87(2):43–50, 1982.

[49] R. Team. Characterization of the Martian Surface Deposits by the Mars Pathfinder Rover, Sojourner. *Science*, 278(5344):1765–1768, December 1997.

[50] Washington University in St. Louis. PGS Geosciences Node Data and Services: MGS MOLA, 2013.

[51] Meng Zou, J.H. Zhang, and L.Q. Ren. Study on the Traction Performance for the Driving wheel of Lunar Rover. *Journal of Astronautics*, 30(1):98–103, 2009.

[52] M Sutoh, K Nagaoka, K Nagatani, and K Yoshida. Design of wheels with grousers for planetary rovers traveling over loose soil. *Journal of Terramechanics*, 2013.

[53] R. Bauer, W. Leung, and T. Barfoot. Experimental and simulation results of wheel-soil interaction for planetary rovers. *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 586–591, 2005.

Appendix A

Wheel-Soil Interaction Analysis

This appendix presents an analysis of wheel-soil interaction forces to determine the net tractive effort that a wheel can generate, based on the components in Table 2.2.

A.1 Soil Thrust

Soil thrust is the primary tractive component that contributes to drawbar pull. Broadly speaking, soil thrust is dependent upon the soil's ability to stick to itself. The Mohr-Coulomb failure criterion considers the soil's "stickiness" to be a combination of two phenomena. The first is the ability of the soil material to hold together without any outside forces, known as cohesion. The second is friction that causes the soil material to hold together under the presence of an outside, normal force on the soil.

Equation A.1 gives the available soil thrust, based on the Mohr-Coulomb criterion, for a wheel that may be slipping in the soil [27, 30]. c, ϕ, and K are properties of the soil. c is the cohesion of the soil. ϕ is the angle of internal shearing resistance of the soil. K is the modulus of shear deformation, which measures the soil displacement required to produce the maximum shear stress in the soil.

$$F_s = (cA_w + W_w \tan\phi)\left(1 - e^{-sl_w/K}\right) \tag{A.1}$$

W_w is the loading on the wheel. s is the wheel slip, defined by Equation A.2. v is the linear velocity of the wheel, r_w is the radius of the wheel, and ω is the angular velocity of the wheel. l_w, shown in Figure A.1, and A_w are the horizontal contact length and horizontal contact area between the wheel and the soil, given by Equations A.3 and A.4.

$$s = 1 - \frac{v}{r_w\omega} \tag{A.2}$$

$$l_w = \sqrt{(d_w - z_w)\, z_w} \tag{A.3}$$

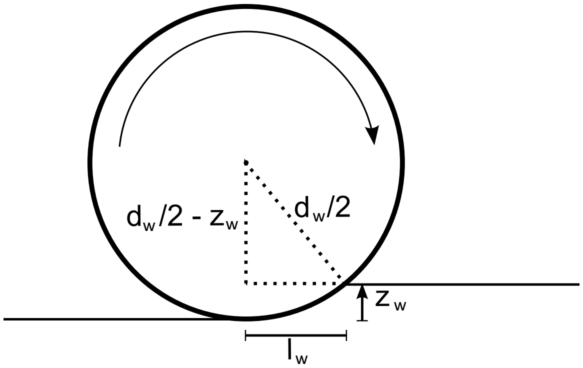


Figure A.1: Wheel contact length in loose soil

$$A_w = b_w l_w \tag{A.4}$$

b_w is the width of the wheel. z_w is the sinkage, the distance that the wheel sinks into the soil, given by Equation A.5 [30]. n is the exponent of soil deformation, k_c is the cohesive modulus of soil deformation, and k_ϕ is the frictional modulus of soil deformation. Each of these is a parameter of the soil. Note that k_c must be in units of force divided by distance raised to the power of $n+1$, and k_ϕ must be in units of force divided by distance raised to the power of $n+2$.

$$z_w = \left[\frac{3W}{(3-n)\left(k_c + b_w k_\phi\right)\sqrt{d_w}}\right]^{\frac{2}{2n+1}} \tag{A.5}$$

A.2 Grouser Thrust

In loose soil grousers provide additional traction via two means [30]. First, if the grousers are close together, then the spaces between them will fill with soil and the diameter of the wheel will effectively be increased. On the other hand, if the grousers are separated enough, they will provide additional traction by acting as a blade that cuts the soil to apply pressure in the horizontal direction.

The robot model in Chapter 2.1 assumes that the robot's wheels have grousers that are separated an appropriate distance to act in the latter case. I set the length of the grousers to 5 percent of the wheel diameter to increase drawbar pull significantly while remaining within practical limits. Some experimental work found that increasing the grouser size continued to improve drawbar pull [31, 33]. Other work found that drawbar pull begins to plateau when grouser lengths reach slightly over 5 percent of the wheel diameter. Additionally, at that size the grousers can begin to have significant negative impacts on other aspects of mobility, such as increasing steering resistance [34].

The effects of grousers on vehicle performance are not completely understood and continue to be studied. While some experimental work has found that grousers can increase the drawbar pull generated by a wheel by 60 percent [31], other experimental work has found that grousers can increase drawbar pull by a factor of 5, but that the effect of grousers is heavily dependent upon the soil conditions [51]. Grouser research also focuses on determining the effect of grouser spacing on mobility performance [33, 32, 52, 53]. Additionally, grousers seem to increase drawbar pull not only by providing additional thrust, but by reducing bulldozing and compaction resistance [33].

This model will account for the effects of grousers by considering them to act as plates pushing parallel to the surface of the soil. Equation A.6 gives the tractive force generated by a grouser of length h_g [30]. N_ϕ is flow value determined by Equation A.7. γ_s is the unit weight of the soil in units of force per volume. W_w, the wheel loading, is used as the surcharge pressure on the soil.

$$F_g = b_w h_g \left(\frac{1}{2} \gamma_s N_\phi + W_w N_\phi + 2c\sqrt{N_\phi} \right) \tag{A.6}$$

$$N_\phi = \tan^2 \left(45^\circ + \frac{\phi}{2} \right) \tag{A.7}$$

A.3 Compaction Resistance

Compaction resistance results from the work that the wheel puts into compressing the soil. Figure A.1 shows how the wheel would depress the soil as it moves forward. Equation A.8 gives the compaction resistance of a rigid wheel in loose soil [30].

$$R_c = \frac{\left(\frac{3W_w}{\sqrt{d_w}} \right)^{\frac{2n+2}{2n+1}}}{(3-n)^{\frac{2n+2}{2n+1}} (n+1) (k_c + b_w k_\phi)^{\frac{1}{2n+1}}} \tag{A.8}$$

n is the exponent of soil deformation, k_c is the cohesive modulus of soil deformation, and k_ϕ is the frictional modulus of soil deformation. Notice that R_c greatly depends on the value of n, which relates the pressure and sinkage for the soil. Soils that require larger amounts of pressure to depress have larger n values.

A.4 Bulldozing Resistance

Bulldozing resistance results from the work put into pushing the soil forward or to the side of the wheel. Equation A.9 gives the bulldozing resistance of a rigid wheel in loose soil [27], which accounts for several modes of failure.

$$R_b = \frac{b_w \sin(\alpha + \phi)}{2 \sin \alpha \cos \phi} \left(2 z_w c K_c + \gamma z_w^2 K_\gamma \right) + \frac{\pi l_r^3 (90 - \phi)}{540} + \frac{\pi c l_r^2}{180} + c l_r^2 \tan \left(45^\circ - \frac{\phi}{2} \right) \tag{A.9}$$

α, l_r, K_c, and K_γ are values related to characteristics of failure for the soil, including failure angles and distances, given by Equations A.10 through A.13. N_c and N_γ are bearing capacity factors developed by Terzaghi. Terzaghi presents charts that provide estimates for the bearing capacity factors based on the angle of internal shearing resistance, ϕ [28, 29].

$$\alpha = \arccos\left(1 - \frac{2z_w}{d_w}\right) \tag{A.10}$$

$$l_r = z_w \tan^2\left(45^\circ - \frac{\phi}{2}\right) \tag{A.11}$$

$$K_c = (N_c - \tan\phi)\cos^2\phi \tag{A.12}$$

$$K_\gamma = \left(\frac{2N_\gamma}{\tan\phi} + 1\right)\cos^2\phi \tag{A.13}$$

A.5 Rolling Resistance

Rolling resistance captures the effects of resistive forces that are internal to the robot, including friction within the robot and wheel deflection. This model accounts for those resistances via a coefficient of rolling friction, f, shown in Equation A.14.

$$R_r = fW_w \tag{A.14}$$

If a designer found it necessary to account for rolling resistance changes across designs, he could perform an analysis to determine how the robot design parameters, such as dimensions, affect the friction coefficient. However, if the same materials are used across designs, the dimensions are unlikely to have a large effect on the friction coefficient, because contact area generally does not affect the magnitude of friction generated.

Appendix B

Robot-Obstacle Interaction Analyses

B.1 Slopes

A stability analysis, shown in Figure B.1, determines the maximum uphill and downhill slopes that the robot can traverse without tipping over. Assuming dynamics are negligible, the maximum stable slope causes the center of gravity of the robot to be directly above the contact point between the wheel and the ground. Equation B.1 shows how the maximum uphill slope angle, θ, can be determined from the dimensions of the robot. B is the robot wheelbase, z_{cg} is the height of the CoG, ϵ is the CoG bias.

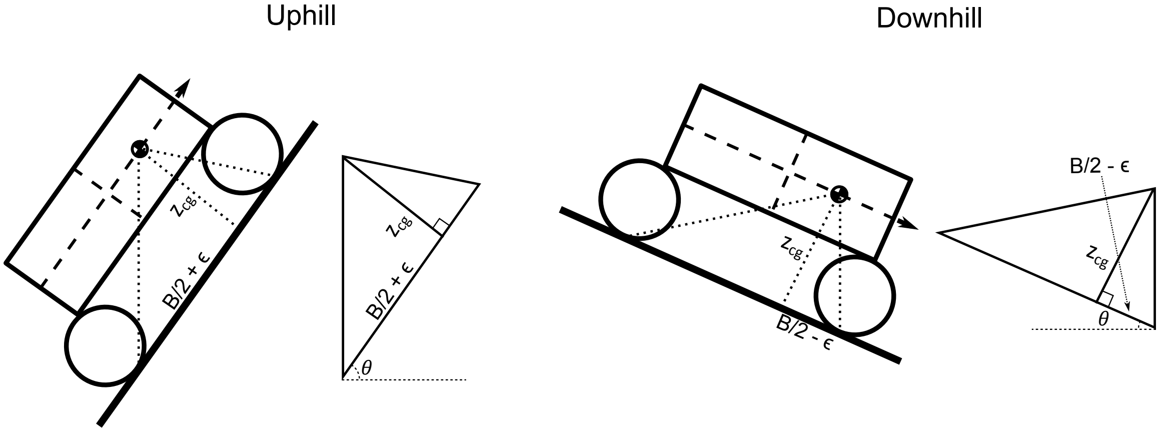


Figure B.1: Geometry for determining slope stability during uphill and downhill traverses

$$\theta_{up}^{stable} = \arctan \frac{B/2 + \epsilon}{z_{cg}} \tag{B.1}$$

Equation B.2 presents the maximum stable downhill slope using the same analysis.

$$\theta_{dn}^{stable} = \arctan \frac{B/2 - \epsilon}{z_{cg}} \tag{B.2}$$

Figure B.2 shows the stability analysis for determining the maximum stable crosshill slope.

Based on this, Equation B.3 is the maximum stable crosshill slope that the robot can traverse.

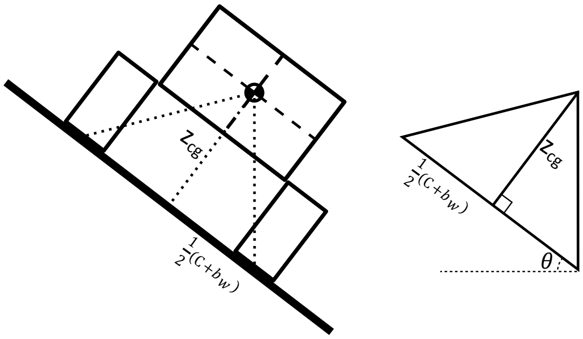


Figure B.2: Geometry for determining crosshill stability

$$\theta_{cs}^{stable} = \arctan \frac{C + b_w}{2 z_{cg}} \tag{B.3}$$

The traction analysis involves determining if enough drawbar pull is generated by the wheel-soil interactions for the robot to maintain its velocity up the slope. This is only a concern during uphill traverses, since during downhill traverses the force of gravity will be assisting the robot. If the robot can generate positive drawbar pull on level ground, in most cases it should be able to generate positive drawbar pull during on any downhill slope.

Figure B.3 shows the forces present during a quasi-static uphill traverse. W is the weight of the robot, F_T is the drawbar pull generated at each wheel, and R is the normal reaction forces from the slope. The front (subscript "1") and rear (subscript "2") reaction forces are presented in Equations B.4 and B.5.

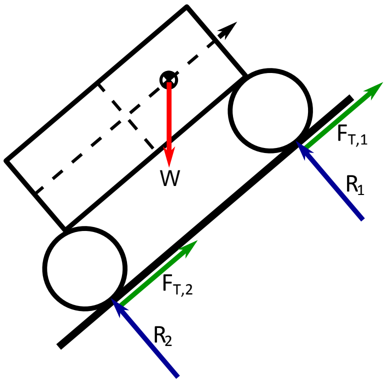


Figure B.3: Traction analysis for a robot traversing an uphill slope

$$R_1 = \frac{W}{B} \left[\left(\frac{B}{2} + \epsilon \right) \cos\theta - z_{cg} \sin\theta \right] \tag{B.4}$$

$$R_2 = \frac{W}{B} \left[\left(\frac{B}{2} - \epsilon \right) \cos\theta + z_{cg} \sin\theta \right] \tag{B.5}$$

The terramechanics model presented in Section 2.3.1 determines the net tractive force produced by each wheel based on their reaction forces. Shown in Equation B.6, if the sum of the tractive forces is greater than the resistive force due to gravity, the robot will be able to traverse the slope.

$$2F_{T,1} + 2F_{T,2} > W \sin\theta \tag{B.6}$$

B.2 Bumps (Positive Obstacles)

The stability analysis determines the largest bump that the robot can surmount without tipping over. Shown in Figure B.4, the robot is stable until its center of gravity is directly above either of its axles. The robot design geometry determines the angle ϕ, as shown in Equation B.7. The limiting bump height, h_{max}, is determined by Equation B.8 based on angle ϕ. As long as the bump height is less than the maximum, shown in Equation B.9, the robot will remain stable while traversing the bump.

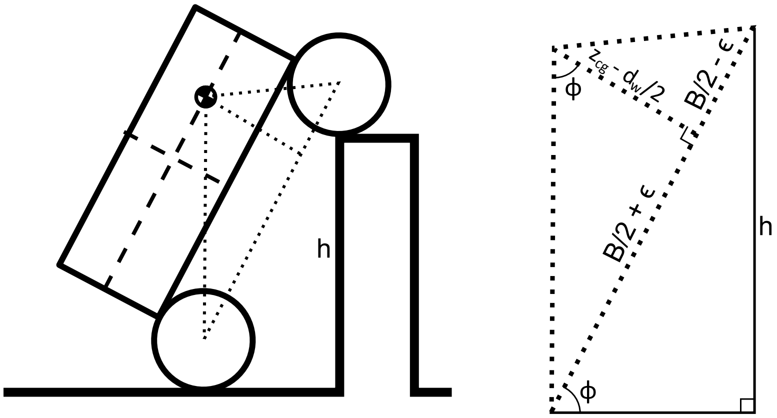


Figure B.4: Robot geometry for determining the maximum bump that can be stably traversed

$$\phi = \arctan \frac{\frac{B}{2} + \epsilon}{z_{cg} - \frac{d_w}{2}} \tag{B.7}$$

$$\frac{h_{max}}{B} = \sin\phi = \frac{\frac{B}{2} + \epsilon}{\sqrt{\left(z_{cg} - \frac{d_w}{2}\right)^2 + \left(\frac{B}{2} + \epsilon\right)^2}} \tag{B.8}$$

$$h < \frac{B\left(\frac{B}{2} + \epsilon\right)}{\sqrt{\left(z_{cg} - \frac{d_w}{2}\right)^2 + \left(\frac{B}{2} + \epsilon\right)^2}} \tag{B.9}$$

The traction analysis determines whether the robot can generate enough drawbar pull to overcome the resistive forces generated by the bump. The traction analysis is performed for bump-climbing with the front wheel and bump-climbing with the rear wheel. To traverse the bump, the robot has to be able to climb it successfully with both front wheels and rear wheels.

Figure B.5 shows the forces on the robot during a bump climb with the front wheel. There are four unknown forces, $F_{T,1}$, R_1, $F_{T,2}$, and R_2, which require four equations to be solved. Equation B.10 presents the force balance in the horizontal direction. Equation B.11 presents the force balance in the vertical direction. Equation B.12 presents the moment balance about the wheel-obstacle contact point.

Lastly, Equation B.13 presents a relationship between the reaction and traction forces at the front wheel. At the threshold point of the bump climb, the front wheel is no longer generating traction from the soil. Instead, it now generates thrust and resistance at the wheel-obstacle interface, based on a Coulombic model of friction. μ is the coefficient of friction which determines the thrust that can be generated. f is the coefficient of rolling resistance. The difference between the two gives the net tractive effort that can be generated, shown in Equation B.13.

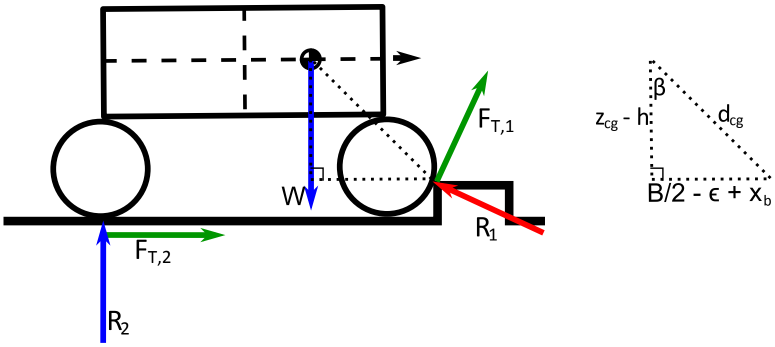


Figure B.5: Forces and dimensions for traction analysis of a robot climbing a bump with its front wheels

$$F_{T,2} - R_1 \cos\alpha + F_{T,1} \sin\alpha = 0 \tag{B.10}$$

$$R_2 + F_{T,1} \cos\alpha + R_1 \sin\alpha - W = 0 \tag{B.11}$$

$$R_2 \left(B + x\right) - W d_{cg} \sin\beta - F_{T,2} h = 0 \tag{B.12}$$

$$F_{T,1} = R_1 \left(\mu - f\right) \tag{B.13}$$

Equations B.10 through B.13 yield the following solution.

$$R_1 = \frac{W \left(B + x_b - d_{cg} \sin\beta\right)}{\left(B + x_b\right)\left(\left(\mu - f\right)\cos\alpha + \sin\alpha\right) + h\left(\cos\alpha + \left(f - \mu\right)\sin\alpha\right)} \tag{B.14}$$

$$F_{T,2} = R_1 \left(\cos\alpha + \left(f - \mu\right)\sin\alpha\right) \tag{B.15}$$

$$R_2 = W - R_1 \left((\mu - f) \cos\alpha + \sin\alpha \right) \tag{B.16}$$

Based on the solution given by Equations B.13 through B.16, the robot can climb the bump if its rear wheels can generate a larger net tractive effort than specified by Equation B.15. This is determined via the terramechanics model (Section 2.3.1) using the reaction force on the rear wheels, R_2.

α, β, x, and d_{cg} are dimensions and angles that can be determined from the robot design and obstacle magnitude. Figure B.6 shows the obstacle contact angle, α, and the obstacle contact distance, x_b. Equations B.17 and B.18 show their calculations. The quantities are the same for both the front wheel traction analysis and rear wheel traction analysis (which will be presented next).

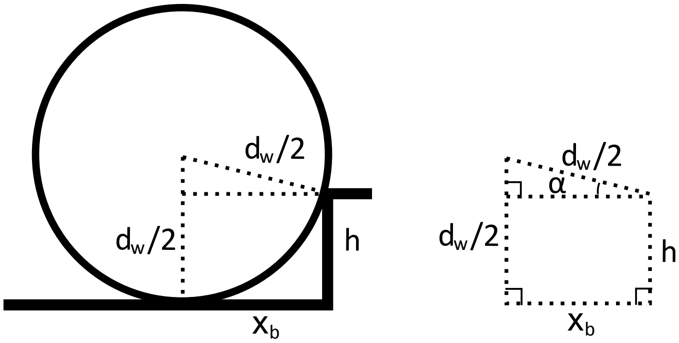


Figure B.6: Geometry of the contact distance and contact angle for a bump

$$x_b = \frac{1}{2}\sqrt{d_w^2 - (d_w - 2h)^2} \tag{B.17}$$

$$\alpha = \arcsin \frac{d_w - 2h}{d_w} \tag{B.18}$$

Figure B.5 shows the center of gravity distance, d_{cg}, and the center of gravity angle, β. Equations B.19 and B.20 show the calculations for these measurements.

$$d_{cg} = \sqrt{\left(\frac{B}{2} + x_b - \epsilon\right)^2 + (z_{cg} - h)^2} \tag{B.19}$$

$$\beta = \arctan \frac{\frac{B}{2} + x_b - \epsilon}{z_{cg} - h} \tag{B.20}$$

The traction analysis for climbing with the rear wheels is similar to that with the front wheels. Figure B.7 shows the forces on the robot while climbing a bump with its rear wheels. Equations B.21 through B.24 present the system of equations for the forces.

$$F_{T,1} + F_{T,2} \sin\alpha - R_2 \cos\alpha = 0 \tag{B.21}$$

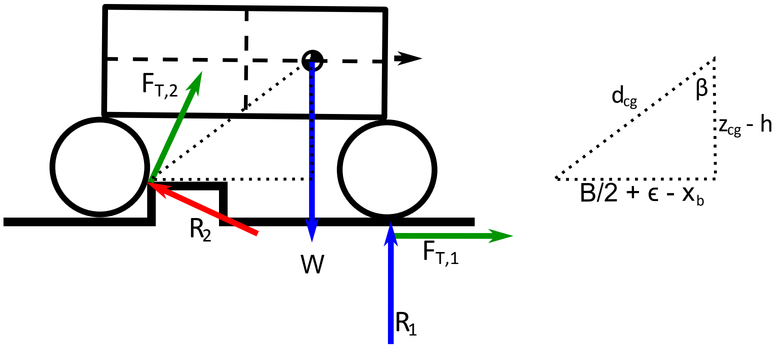


Figure B.7: Forces and dimensions for traction analysis of bump climb with front wheels

$$R1 + F_{T,2} \cos \alpha - R_2 \sin \alpha - W = 0 \tag{B.22}$$

$$W d_{cg} \sin \beta - F_1 h - R_1 \left(B - x_b\right) = 0 \tag{B.23}$$

$$F_{T,2} = R_2 \left(\mu - f\right) \tag{B.24}$$

The solution to these equations is presented below.

$$R_2 = \frac{W \left(B - x_b + h - d_{cg} \sin \beta\right)}{\left(B - x_b\right) \left(\left(\mu - f\right) \cos \alpha + \sin \alpha\right) - h \left(\cos \alpha + \left(f - \mu\right) \sin \alpha\right)} \tag{B.25}$$

$$F_{T,1} = R_2 \left(\cos \alpha + \left(f - \mu \sin \alpha\right)\right) \tag{B.26}$$

$$R_1 = W - R_2 \left(\left(\mu - f\right) \cos \alpha + \sin \alpha\right) \tag{B.27}$$

Similar to the previous analysis, if the net tractive effort from the front wheel, as determined by the terramechanics model, is greater than the required force specified by Equation B.26, then the robot will have enough traction to traverse the obstacle. Notice that the CoG distance, d_{cg}, and CoG angle, β, are different than in the case of the front-wheel bump climb. This is evident from comparing Figure B.5 and Figure B.7. Equations B.28 and B.29 show the calculations for these measurements for the rear-wheel climb.

$$d_{cg} = \sqrt{\left(\frac{B}{2} - x_b + \epsilon\right)^2 + \left(z_{cg} - h\right)^2} \tag{B.28}$$

$$\beta = \arctan \frac{\frac{B}{2} - x_b + \epsilon}{z_{cg} - h} \tag{B.29}$$

The geometric analysis determines the largest bump that the robot can surmount without having the obstacle collide with its body. This model conservatively assumes that the length of the bump is shorter than the distance between the front and rear wheels. Because of this, the body of the robot will collide with the obstacle unless the robot's clearance (equal to its wheel diameter) is greater than the size of the bump:

$$d_w \geq h \tag{B.30}$$

B.3 Ditches (Negative Obstacles)

Figure B.8 shows the free body diagram of the robot traversing a ditch with its front wheels. As with the bump, the four unknown forces can be determined by force and moment balances and by a friction relationship at the wheel-ditch interface. Equations B.31 through B.34 show these relationships.

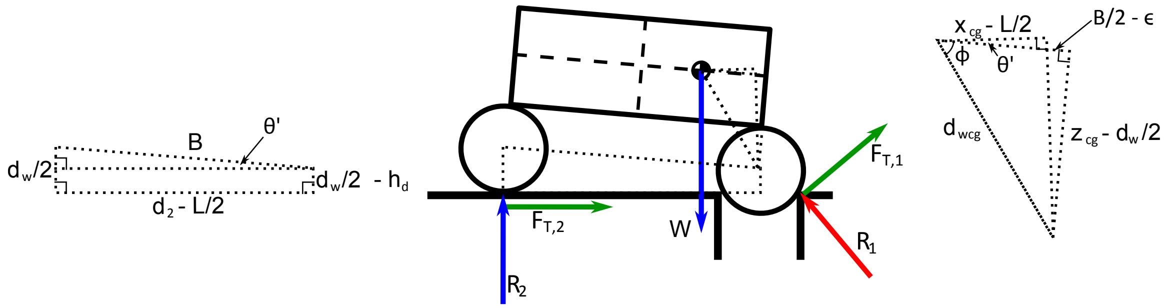


Figure B.8: Forces and geometries for traction analysis of a robot climbing a ditch obstacle with its front wheels

$$F_{T,2} + F_{T,1}\sin\alpha - R_1\cos\alpha = 0 \tag{B.31}$$

$$R_2 + F_{T,1}\cos\alpha + R_1\sin\alpha - W = 0 \tag{B.32}$$

$$Wx_{cg} - R_2 d_2 = 0 \tag{B.33}$$

$$F_{T,1} = R_1\left(\mu - f\right) \tag{B.34}$$

Solving for the forces yields Equations B.35 through B.37.

$$R_1 = \frac{W\left(1 - \frac{x_{cg}}{d_2}\right)}{\left(\mu - f\right)\cos\alpha + \sin\alpha} \tag{B.35}$$

$$F_{T,2} = R_1\left(\cos\alpha - \left(\mu - f\right)\sin\alpha\right) \tag{B.36}$$

$$R_2 = W\frac{x_{cg}}{d_2} \tag{B.37}$$

The robot is able to generate enough traction to traverse the ditch if the maximum net tractive effort of the rear wheels is greater than the force specified by Equation B.36. The terramechanics model provides the maximum net tractive effort based off of the wheel loading, R_2, given by Equation B.37.

d_2, x_{cg}, and α are dimensions and angles that can be determined based on the robot's design and the magnitude of the ditch. Figure B.9 shows the geometry of the contact between the wheel and the ditch. Equation B.38 gives the contact angle, α. Equation B.39 gives the contact height, h_d. For both equations L is the length of the ditch.

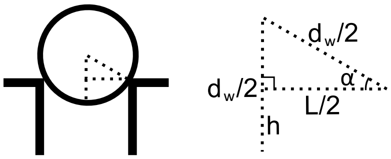


Figure B.9: Geometry of the contact angle and contact height for a ditch

$$\alpha = \arccos\frac{L}{d_w} \tag{B.38}$$

$$h_d = \frac{1}{2}\left(d_w - \sqrt{d_w^2 - L^2}\right) \tag{B.39}$$

Figure B.8 shows d_2, the distance from the rear-wheel contact point to the point where the front wheel contacts the far edge of the ditch, along with lengths d_{wcg} and x_{cg} and the angle the robot pitches forward, θ'. Equation B.40 shows the calculation for d_2, and Equation B.41 shows the calculation for θ'. The other two lengths are shown in the two overlapping triangles on the right of Figure B.8. d_{wcg} is the distance from the center of gravity to the axle of the wheel in contact with the ditch (Equation B.42). x_{cg} is the horizontal distance from the CoG to the point where the wheel contacts the ditch (Equation B.43).

$$d_2 = \sqrt{B^2 - h_d^2} + \frac{L}{2} \tag{B.40}$$

$$\theta' = \arcsin\frac{h}{B} \tag{B.41}$$

$$d_{wcg} = \sqrt{\left(z_{cg} - \frac{d_w}{2}\right)^2 + \left(\frac{B}{2} - \epsilon\right)^2} \tag{B.42}$$

$$x_{cg} = d_{wcg}\cos\left(\theta' + \phi\right) + \frac{L}{2} \tag{B.43}$$

Figure B.10 shows the free body diagram of a robot climbing a ditch obstacle with its rear wheel. Equations B.44 through B.47 show the force and moment balances.

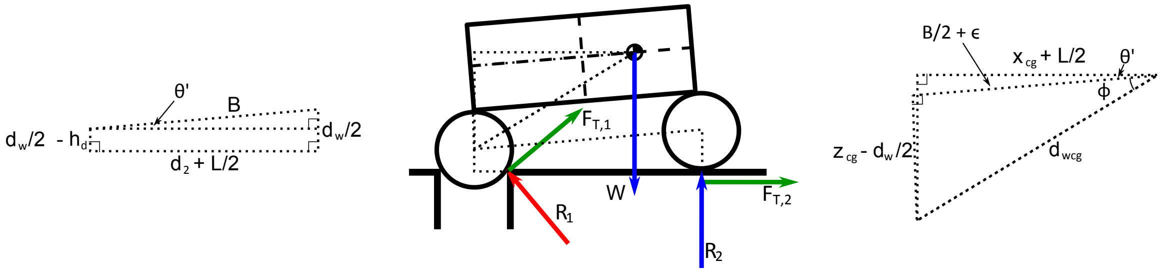


Figure B.10: Forces and geometries for traction analysis of a robot climbing a ditch obstacle with its rear wheels

$$F_{T,1} + F_{T,2}\sin\alpha - R_2\cos\alpha = 0 \tag{B.44}$$

$$R_1 + F_{T,2}\cos\alpha + R_2\sin\alpha - W = 0 \tag{B.45}$$

$$W x_{cg} - R_1 d_1 = 0 \tag{B.46}$$

$$F_{T,2} = R_2\left(\mu - f\right) \tag{B.47}$$

Solving for the forces yields Equation B.48 through B.50:

$$R_2 = \frac{W\left(1 - \frac{x_{cg}}{d_1}\right)}{\left(\mu - f\right)\cos\alpha + \sin\alpha} \tag{B.48}$$

$$F_{T,1} = R_2\left(\cos\alpha - \left(\mu - f\right)\sin\alpha\right) \tag{B.49}$$

$$R_1 = W\frac{x_{cg}}{d_1} \tag{B.50}$$

As before, the maximum tractive effort from the front wheels, as determined by the terramechanics model, must be greater than that given by Equation B.49.

The obstacle contact angle, α, ditch contact height, h_d, and robot pitch angle, θ' are the same as those for the front wheel ditch analysis (Equations B.38, B.39, and B.41). The other dimensions are different if the center of gravity is off-center front to rear in the robot.

Equation B.51 gives d_1, the distance from the front-wheel contact point to the point where the rear wheel contacts the near edge of the ditch. Equation B.52 gives, d_{wcg}, the distance from the CoG to the rear axle. Lastly, Equation B.53 gives x_{cg}, the horizontal distance between the CoG and the rear-wheel ditch contact point.

$$d_1 = \sqrt{B^2 - h_d^2} - \frac{L}{2} \tag{B.51}$$

$$d_{wcg} = \sqrt{\left(z_{cg} - \frac{d_w}{2}\right)^2 + \left(\frac{B}{2} + \epsilon\right)^2} \tag{B.52}$$

$$x_{cg} = d_{wcg} \cos\left(\theta' + \phi\right) - \frac{L}{2} \tag{B.53}$$

The geometric analysis for this model checks if the robot can traverse the ditch without its body colliding with any part of the ditch. The only circumstance under which there can be a collision is if the length of the ditch is greater than the diameter of the wheel. Equation B.54 gives the circumstance under which the robot will avoid a collision.

$$d_w > L \tag{B.54}$$

Appendix C

Robot Chassis Stress Analyses

The robot chassis must be strong enough to resist yielding from stresses generated by any terrain features. This model determines the maximum stress imposed on the chassis by each type of obstacle through a quasi-static analysis. The chassis is assumed to be a beam, with the maximum stress determined based on the reaction forces at the wheel and the payload weight of the robot concentrated at the location of the CoG. The model could consider the payload of the robot to be distributed across the length of the chassis. However, if this is the case then the maximum stress in the robot decreases as the length of the robot increases. Using a point mass is a conservative assumption and avoids this problematic trend.

The maximum stress in the robot is the sum of the maximum bending stress and maximum axial stress, as shown in Equation C.1.

$$\sigma_{max} = \sigma_{bend} + \sigma_{axial} \tag{C.1}$$

Due to the weight of the robot payload, the axial stress in the front and rear portions of the chassis will be different. The axial stress used in Equation C.1 is the larger of the front axial stress, σ_{front}, and the rear axial stress, σ_{rear}. The bending stress, given by Equation C.2, depends on the maximum bending moment in the chassis, M_{max}. Note that the bending stress during crosshill travel must be calculated in different manner.

$$\sigma_{bend} = \frac{6M_{max}}{b_c h_c^2} \tag{C.2}$$

C.1 Uphill and Downhill Stress

Figure C.1 shows how the wheel forces and payload weight transfer to the robot chassis during uphill travel. Because this is a quasi-static analysis, at each wheel the normal force, R, and the traction, F_T, are dependent. The component of robot weight on the front (W_f) or rear (W_r) wheels is determined by just R or F_T, as shown in Equations C.3 and C.4. The analysis in Section 2.3.2 provides the wheel forces.

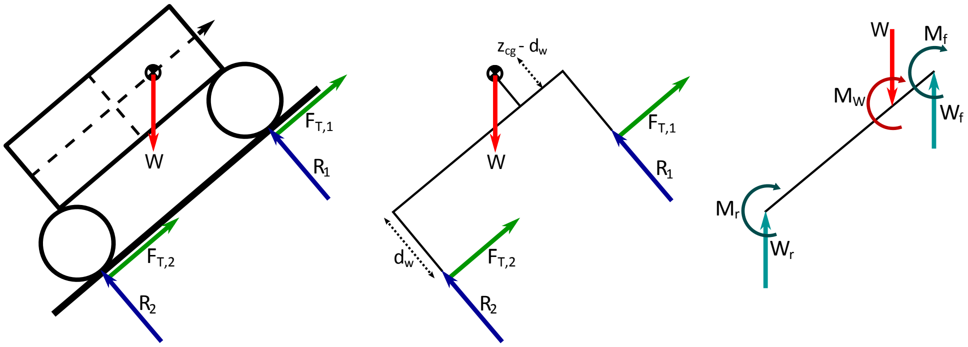


Figure C.1: Analysis of forces and moments on the robot chassis during an uphill traverse

$$W_f = \frac{R_1}{\cos\theta} \tag{C.3}$$

$$W_r = \frac{R_2}{\cos\theta} \tag{C.4}$$

Equation C.5 and Equation C.6 give the axial stress in the front and rear portions of the chassis.

$$\sigma_{front} = \frac{W_f \sin\theta}{b_c h_c} \tag{C.5}$$

$$\sigma_{rear} = \frac{W_r \sin\theta}{b_c h_c} \tag{C.6}$$

The bending moments on the chassis are determined from the angle of the slope and the front and rear wheel forces, as shown in Equations C.7 though C.9.

$$M_w = \frac{W}{2}\left(z_{cg} - d_w\right)\sin\theta \tag{C.7}$$

$$M_f = R_1 d_w \tan\theta \tag{C.8}$$

$$M_r = R_2 d_w \tan\theta \tag{C.9}$$

The maximum bending moment, given by Equation C.10, occurs at the point on the chassis where the payload mass is applied.

$$M_{max} = M_f + W_f\left(\frac{B}{2} - \epsilon\right) + M_w \tag{C.10}$$

C.2 Crosshill Stress

During crosshill travel, one side of the ladder frame chassis bears more than half of the robot's mass. The rungs on the ladder frame share a combined axial and bending stresses, similar to the analysis in Section C.1. The model assumes there are four rungs and that they are the same size as the longitudinal beams. Thus, the stress in the longitudinal beams will generally be larger.

Figure C.2 shows the robot traveling crosshill with both the front to back and left to right weight distributions. Equation C.11 gives the component of weight on the right side of the chassis. Equation C.12 gives the component of weight on the front-right wheel.

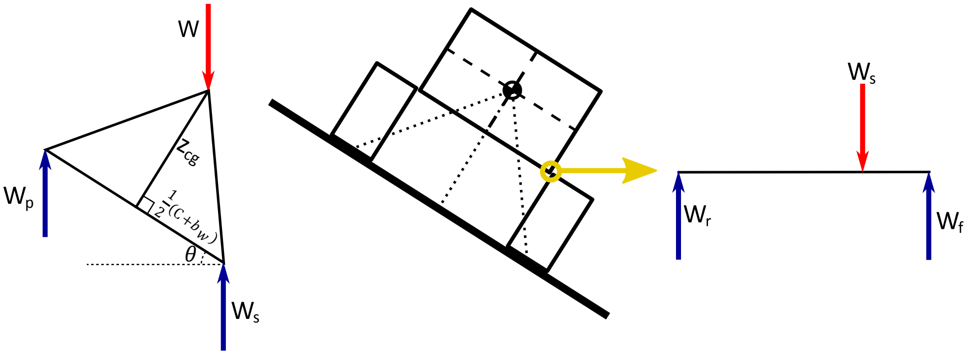


Figure C.2: Forces on the robot chassis during crosshill travel, including a front view (left) and side view (right)

$$W_s = W\left(\frac{1}{2} + \frac{z_{cg}}{C + b_w}\tan\theta\right) \tag{C.11}$$

$$W_f = \frac{W_s}{B} = \frac{W}{B}\left(\frac{1}{2} + \frac{z_{cg}}{C + b_w}\tan\theta\right) \tag{C.12}$$

The maximum bending moment in the chassis is given by Equation C.13.

$$M_{max} = W_f\left(\frac{B}{2} - \epsilon\right) \tag{C.13}$$

Because the robot is tilted, the forces on the chassis are diagonal across the rectangular chassis cross section. The maximum stress can be determined from the sum of the component forces parallel to the sides of the cross section. This method does not use Equation C.2 to determine the stress. Instead, Equations C.14 and C.15 give the horizontal and vertical bending stress components relative to the cross section. Equation C.16 gives the total bending stress during crosshill travel.

$$\sigma_h = \frac{6M_{max}\cos\theta}{b_c h_c^2} \tag{C.14}$$

$$\sigma_v = \frac{6M_{max}\sin\theta}{b_c^2 h_c} \tag{C.15}$$

$$\sigma_{max} = 6W_f \left(\frac{B}{2} - \epsilon\right)\left(\frac{\cos\theta}{b_c h_c^2} + \frac{\sin\theta}{b_c^2 h_c}\right) \tag{C.16}$$

C.3 Bump Stress

To determine the maximum stress on the chassis from a bump obstacle, the stress must be determined for the robot climbing the bump with its front wheels and for the robot climbing with its rear wheels. The larger of those stresses is the maximum stress from a bump obstacle.

Figure C.3 shows the forces on the chassis during a front-wheel climb of the obstacle. The analysis in Section 2.3.3 determines the wheel forces. M_w is given by Equation C.7, with θ equal to zero. Equations C.17 and C.18 give M_f and W_f, which are determined from the geometry of the forces on the wheels.

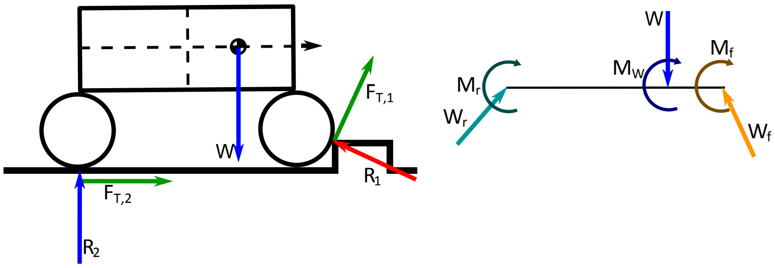


Figure C.3: Forces on the robot's chassis while it is traversing a bump with its front wheels

$$M_f = \frac{d_w}{2} F_{T,1} \tag{C.17}$$

$$W_f = \sqrt{R_1^2 + F_{T,1}^2} \tag{C.18}$$

Equation C.19 gives β_1, the angle between the contact angle, α (given by Equation B.18), and the force vector between the front wheel and chassis. β_2 is the angle between the vertical and the force vector between the rear wheel and the chassis.

$$\beta_1 = \arctan \frac{F_{T,1}}{R_1} \tag{C.19}$$

$$\beta_2 = \arctan \frac{F_{T,2}}{R_2} \tag{C.20}$$

These angles, which give the axial force components, are used in Equations C.21 and C.22 to determine the axial stress in the front and rear portions of the chassis.

$$\sigma_{front} = \frac{W_f \cos(\alpha + \beta_1)}{bh} \tag{C.21}$$

$$\sigma_{front} = \frac{W_r \sin\beta_2}{bh} \tag{C.22}$$

Equation C.23 gives the maximum bending moment, which occurs at the payload loading point on the chassis.

$$M_{max} = M_f + W_f\left(\frac{B}{2} - \epsilon\right)\sin(\alpha + \beta_1) + M_w \tag{C.23}$$

Figure C.4 shows the forces applied to the chassis while the robot climbs a bump with its rear wheels. These forces and moments can be determined in the same manner as with the front-wheel climb, using Equations C.17 through C.20. Similarly, Equation C.7, with θ set to zero, gives M_w.

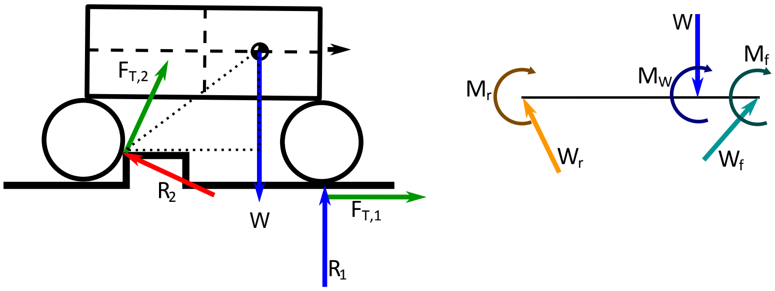


Figure C.4: Chassis forces while the robot is traversing a bump with its rear wheels

Equations C.24 through C.26 give the axial stresses and the maximum bending moment.

$$\sigma_{front} = \frac{W_f \sin\beta_1}{bh} \tag{C.24}$$

$$\sigma_{rear} = \frac{W_r \cos(\alpha + \beta_2)}{bh} \tag{C.25}$$

$$M_{max} = M_f + W_f\left(\frac{B}{2} - \epsilon\right)\cos\beta_1 + M_w \tag{C.26}$$

C.4 Ditch Stress

The stress analysis for a ditch traverse is similar to that for a bump traverse, with differences arising from the tilt of the robot. For a ditch, the instant of maximum obstacle resistance occurs when the robot's wheel is in contact with both sides of the ditch. At this point the robot's chassis is tilted relative to the ground. Equation B.41 gives that angle, θ'.

Figure C.5 shows the forces on the robot chassis while the front wheels traverse of a ditch. The wheel forces are given by the analysis in Appendix B.3. The chassis forces, as well as the angles β_1 and β_2, can be determined via the same analysis in Section C.3, using $\theta = \theta'$.

The axial stresses and the maximum bending moment are given by Equations C.27 through C.29.

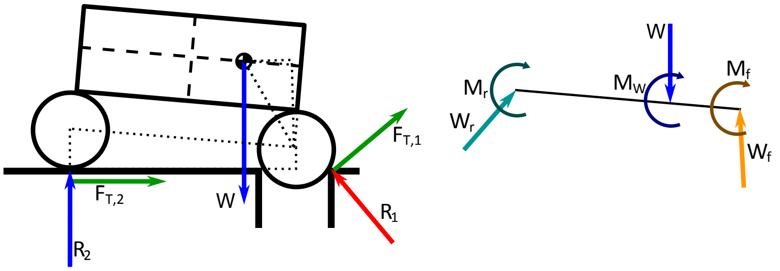


Figure C.5: Forces on the chassis from the robot crossing a ditch obstacle with its front wheels

$$\sigma_{front} = \frac{W_f \cos\left(\alpha + \beta_1 - \theta'\right)}{bh} \tag{C.27}$$

$$\sigma_{rear} = \frac{W_r \sin\left(\beta_2 - \theta'\right)}{bh} \tag{C.28}$$

$$M_{max} = M_f + W_f \left(\frac{B}{2} - \epsilon\right) \sin\left(\alpha + \beta_1 - \theta'\right) + M_w \tag{C.29}$$

Figure C.6 shows the forces on the robot chassis while the rear wheels traverse the ditch. Appendix B.3 gives the wheel forces and Appendix C.3 gives the chassis forces and angles. The axial stresses and the maximum bending moment are given by Equations C.30 through C.32.

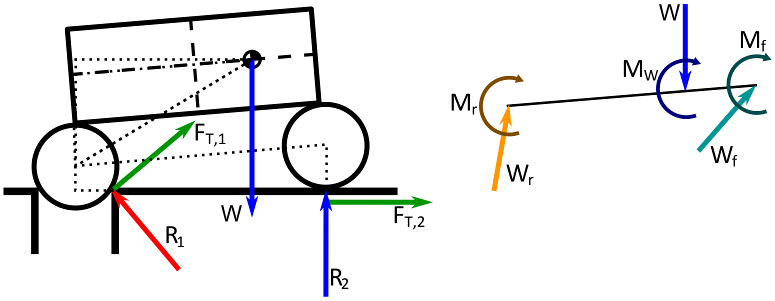


Figure C.6: Forces on the chassis from the robot crossing a ditch obstacle with its rear wheels

$$\sigma_{front} = \frac{W_f \sin\left(\beta_1 + \theta'\right)}{bh} \tag{C.30}$$

$$\sigma_{rear} = \frac{W_r \cos\left(\alpha + \beta_2 + \theta'\right)}{bh} \tag{C.31}$$

$$M_{max} = M_f + W_f \left(\frac{B}{2} - \epsilon\right) \sin\left(\beta_1 + \theta'\right) + M_w \tag{C.32}$$

Appendix D

Cascading Performance Calculation

Robot performance could be measured by either the time, T, required to travel a required distance, $\bar{d}$, or the equivalent straight-line velocity, v_{esl}, of the robot. Upon initial consideration, Equation D.1 and Equation D.2 seem to provide these measures, with the penalty times, $\hat{T}$, as a function of the distance, $\bar{d}$. $\bar{T}$ is the time that the robot would take to travel $\bar{d}$ if there were no obstacles (slopes, bumps or ditches) present. Table 3.3 lists the subscript symbols for $\hat{T}$. Note that T and $\hat{T}$ are random variables, meaning that their exact value cannot be known *a priori*.

$$T = \bar{T} + \hat{T}_{bc} + \hat{T}_{bm} + \hat{T}_{dc} + \hat{T}_{dm} + \hat{T}_{usc} + \hat{T}_{usm} + \hat{T}_{dsc} + \hat{T}_{dsm} \tag{D.1}$$

$$v_{esl} = \frac{\bar{d}}{E\left[T\right]} \tag{D.2}$$

However, the robot will likely be forced to maneuver around some obstacles, resulting in an actual distance traveled, d, that is larger than $\bar{d}$. Thus, the penalty times should be based on d, not $\bar{d}$.

Consider this perspective: along with the time penalties, $\hat{T}_1$, from obstacles encountered along distance $\bar{d}$, there is a additional distance $\hat{d}_1$ that the robot will travel. Along $\hat{d}_1$ there should also be the potential for the robot to encounter obstacles, which would add both additional time delays, $\hat{T}_2$ and additional distance, $\hat{d}_2$. Likewise, the robot can encounter additional obstacles along $\hat{d}_2$, and so on. Thus, to correctly determine T, all of the additional time penalties at each layer must be included as in Equation D.3. Each i will be referred to as a “layer.”

$$T = \bar{T} + \sum_{i=1}^{\infty} \hat{T}_i \tag{D.3}$$

Because T is nondeterministic, its expected value, $E\left[T\right]$, provides a useful single-value measure of performance. Equation D.4 shows how to determine $E\left[T\right]$ by using the expected value of each variable. Note that $\bar{T}$ is deterministic.

$$E\left[T\right] = \bar{T} + \sum_{i=1}^{\infty} E\left[\hat{T}_i\right] \tag{D.4}$$

The summation in Equation D.4 can be split into sets of summations, one for penalties from obstacle climbing, subscript c, and one for penalties from obstacle avoidance, subscript m:

$$E\left[T\right] = \bar{T} + \sum_{i=1}^{\infty} E\left[\hat{T}_{c,i}\right] + \sum_{i=1}^{\infty} E\left[\hat{T}_{m,i}\right] \tag{D.5}$$

Two realizations allow Equation D.5 to be simplified. First, take note that additional distance only results from obstacle avoidance, since the surface path length is not changed by obstacle climbing. Second, the ratios of 1) the time penalties in one layer, i, to 2) the time penalties from avoiding in the previous layer, $i-1$, are equal. This is true for all numerator time penalties due to climbing and for those due to avoiding. Additionally, as presented in Equation D.6 and Equation D.7, this equivalency of these ratios also hold true for the first layer, $i = 1$, where the previous time is $\bar{T}$.

$$K_c = \frac{E\left[\hat{T}_{c,i}\right]}{E\left[\hat{T}_{m,i-1}\right]} = \frac{E\left[\hat{T}_{c,1}\right]}{\bar{T}} \tag{D.6}$$

$$K_m = \frac{E\left[\hat{T}_{m,i}\right]}{E\left[\hat{T}_{m,i-1}\right]} = \frac{E\left[\hat{T}_{m,1}\right]}{\bar{T}} \tag{D.7}$$

Fully substituting K_c and K_m into Equation D.5 at each layer, as demonstrated in Equations D.8 to D.10 yields Equation D.11.

$$E\left[T\right] = \bar{T} + \sum_{i=1}^{\infty} E\left[\hat{T}_{c,i}\right] + \sum_{i=1}^{\infty} E\left[\hat{T}_{m,i}\right] \tag{D.8}$$

$$\begin{aligned} E\left[T\right] =& \bar{T} \quad + \hat{T}_{c,1} \quad + \hat{T}_{m,1} \quad + \hat{T}_{c,2} \quad + \hat{T}_{m,2} \quad + \hat{T}_{c,3} \quad + \hat{T}_{m,3} \quad + \ldots \\ =& \bar{T} \quad + \bar{T}K_c \quad + \bar{T}K_m \quad + \hat{T}_{m,1}K_c \quad + \hat{T}_{m,1}K_m \quad + \hat{T}_{m,2}K_c \quad + \hat{T}_{m,2}K_m \quad + \ldots \\ =& \bar{T} \quad + \bar{T}K_c \quad + \bar{T}K_m \quad + \bar{T}K_mK_c \quad + \bar{T}K_mK_m \quad + \bar{T}K_m^2K_c \quad + \bar{T}K_m^2K_m \quad + \ldots \end{aligned} \tag{D.9}$$

$$\begin{aligned} E\left[T\right] &= \bar{T} + \left[\bar{T}K_c + \bar{T}K_mK_c + \bar{T}K_m^2K_c + \ldots\right] + \left[\bar{T}K_m + \bar{T}K_m^2 + \bar{T}K_m^3 + \ldots\right] \\ &= \left[\bar{T}K_c + \bar{T}K_mK_c + \bar{T}K_m^2K_c + \ldots\right] + \left[\bar{T} + \bar{T}K_m + \bar{T}K_m^2 + \bar{T}K_m^3 + \ldots\right] \\ &= \bar{T}K_c\sum_{i=0}^{\infty} K_m^i + \bar{T}\sum_{i=0}^{\infty} K_m^i \end{aligned} \tag{D.10}$$

$$E\left[T\right] = \left(\bar{T}K_c + \bar{T}\right)\sum_{i=0}^{\infty} K_m^i \tag{D.11}$$

The summation in Equation D.11 is a geometric series, and since $K_m < 1$, it converges as shown to $1/\left(1 - K_m\right)$. Thus, the expected value of the time is:

$$E\left[T\right] = \bar{T}\frac{1 + K_c}{1 - K_m} \tag{D.12}$$

The expected value of the equivalent straight line velocity can be written in terms of the nominal robot velocity, v:

$$v_{esl} = \frac{\bar{d}}{E\left[T\right]} = \frac{\bar{d}}{\bar{T}}\frac{1 - K_m}{1 + K_c} = v\frac{1 - K_m}{1 + K_c} \tag{D.13}$$

Appendix E

Design Constants and Relationships for the Mars Mission Case Study

Table E.1 lists the values of constants and other relationships for the Mars case study in Chapter 4.

Table E.1: Mars case study constants and relationships

Parameter	Symbol	Value
Mass Budget	m_{budget}	1000 lbs
Payload Mass	$m_{payload}$	30 lbs
Electronics Mass	$m_{electronics}$	10 lbs
Thermal System Mass	$m_{thermal}$	5 lbs
Gravity		$12.2\,\frac{\text{ft}}{\text{s}^2}$
Nominal Wheel Slip	s	0.35
Grouser Height		$0.05d_w$
Rolling Friction Coefficient	f	0.1
Obstacle Friction Coefficient	μ	0.5
Chassis Material Yield Strength	σ_{yield}	50 ksi
Chassis Material Density	ρ	$0.1\,\frac{\text{lbm}}{\text{in}^3}$
Nominal Robot Velocity	v	$2\,(2h+L)$
Electrical Power Usage		50 watts
Battery Specific Energy	c_{batt}	$145\,\frac{\text{W-hr}}{\text{kg}}$
Battery backup Time	T_{backup}	3 hours
Solar Irradiance	I	$1300\,\frac{\text{W}}{\text{m}^2}$
Solar Panel Efficiency	η	0.28
Solar Panel Oversizing	s	0.25
Solar Panel Density	ρ_{solar}	$1.87\,\frac{\text{lbf}}{\text{m}^2}$
Power Managment Coefficient	k_P	$38.4\,\frac{\text{lbf}}{\text{m}^2}$
Minimum Wheel Separation	s_w	5 in
Robot Body Height		6 in
Wheel Ratio	r_w	0.1
Chassis Thickness Ratio	r_t	4
Maximum Sinkage	r_s	$0.1d_w$