

Final Report to NASA Lewis Research Center *FINAL*

for Contract NCC3-297

1N-37-CR

by

O C I T

Carnegie Mellon University

207565

Robotics Institute

P-132

Pittsburgh, PA 15213

William Whittaker - Principal Investigator

Kevin Dowling - Project Scientist

March 4, 1994

This is the final report for an equipment contract for NASA Lewis Research Center. We have utilized the funding to implement a walking robot for applications of planetary exploration. As part of this report we have included the following:

- Design Document
- Paper on Daedalus Walking Robot
- Paper on Drivetrain Configuration for a Walking Robot

A model of the walker was completed and delivered to NASA Lewis and the Daedalus Walker itself is nearing completion at this time.

(NASA-CR-195200) [DESIGN OF A
WALKING ROBOT] Final Report
(Carnegie-Mellon Univ.) 132 p

N94-25274

Unclass

G3/37 0207565

433826

1B3p

Daedalus: A Walking Robot for Autonomous Planetary Exploration

Gerald P. Roston
Kevin Dowling

Field Robotics Center
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract:

Carnegie Mellon University's Autonomous Planetary Exploration Program (APEX) is currently building the Daedalus robot; a system capable of performing extended autonomous planetary exploration missions. Extended autonomy is an important capability because the continued exploration of the moon, Mars and other solid bodies within the solar system will probably be carried out by autonomous robotic systems. There are a number of reasons for this - the most important of which are the high cost of placing a man in space, the high risk associated with human exploration and communication delays that make teleoperation infeasible.

The Daedalus robot represents an evolutionary approach to robot mechanism design and software system architecture. Daedalus incorporates key features from a number of predecessor systems. Using previously proven technologies, the Apex project endeavors to encompass all of the capabilities necessary for robust planetary exploration.

1 Introduction

Carnegie Mellon University's Autonomous Planetary Exploration Program (APEX) is currently building the Daedalus robot; a system capable of performing extended autonomous planetary exploration missions. Extended autonomy is an important capability because the initial exploration of the moon, Mars and other solid bodies within the solar system will probably be carried out by autonomous robotic systems. There are a number of reasons for this - the most important of which are the high cost of placing a man in space, the high risk associated with human exploration and communication delays that make teleoperation infeasible.

Many robotic system for planetary exploration have been previously developed. Most of these systems focused on, and adequately demonstrated, various subsets of the capabilities required for autonomous planetary exploration. However, no project has developed all of the required skills simultaneously. The Apex project endeavors to encompass all of the capabilities necessary for robust planetary exploration.

The Daedalus robot represents an evolutionary approach to robot design and incorporates key features from a number of predecessor systems, such as the CMU Ambler, the Martin Marietta frame-walker and others. Among other features, Daedalus combines the Ambler-derived orthogonal-leg design and the Martin walking-beam concept. Using previously proven technologies ensures that the required goals of reliability, terrainability and space relevance will be achieved.

Daedalus' software systems also represents an evolutionary approach that draws from previous works. Unlike the Ambler, which used a strictly hierarchical planning scheme, and unlike the JPL mini-rovers, which use a strictly reactive planning scheme, Daedalus will incorporate a hybrid scheme that draws from the strengths of these two approaches. Foot placement and the basic gait will be primarily reactive in nature, but other modules will involve deliberative planning for missions.

In the course of developing Daedalus, a number of issues were highlighted and resolved. These issues include the ability to space-qualify the robotic system, to design a power and mass efficient robot for carrying out scientific experiments, to economically deliver the robotic system on-board a commercial launch vehicle, to develop robust software capable of functioning for periods of weeks, to develop a system capable of stand-alone exploration missions and to enable planetary exploration by providing a general framework for autonomous mission planning.

A robot, such as Daedalus, is a tightly integrated system comprised of several components: mechanism, computing, sensing, software, etc. This paper will focus specifically on the mechanical design of the Daedalus robot. Section 2 reviews the mission requirements for a lunar exploration mission and discusses how these requirements can be used to make Earth-based missions more space relevant. Section 3 shows how the mission

Copyright © 1993 by Gerald P. Roston and Kevin Dowling. Published by the American Institute of Aeronautics and Astronautics, Inc. with permission.

Mr. Roston is working towards a PhD in Mechanical Engineering.
Mr. Dowling is a Project Scientist with the Robotics Institute.

requirements are used to generate a preliminary design concept for a planetary exploration robot. Some of the mechanical design details of the Daedalus robot are presented in Section 4.

2 Mission Overview

The Daedalus configuration is designed to accommodate two missions, an extended duration lunar mission and a long duration Earth mission. The purpose of the Earth mission is to develop a system capable of performing long-duration, autonomous planetary exploration. Development for the Earth mission has commenced. Although the Daedalus robot itself is not space-qualified, only those component systems that are potentially space qualifiable are utilized.

2.1 Lunar Mission

Lunar mission goals include the exploration of the lunar surface, performing lunar surface scientific experiments, site certification for follow-on missions and exploration of interesting formations such as volcanic vents, impact craters and lava tubes. Unless the lunar rover has the capability of storing large amounts of energy, or unless it possesses radioactive heat sources, the longest probable mission will last one lunar day (14 terran days) since the cold night temperature may damage certain system components. During the course of this mission, the rover is expected to cover upwards of 100 km over a variety of terrains.

2.2 Earth Mission

Earth-based missions serve as analogs for lunar missions by simulating the operating conditions, terrains and interactions. Daedalus will be tested in the south-western US desert because the extreme ruggedness of the terrain is similar to that found on the moon. Candidate sites include Death Valley, CA; Kelso, CA and Cinder Lake, AZ. The goal of the earth mission is a two week, autonomous traverse of a region while performing selected scientific experiments. During this mission, every effort will be taken to simulate the actual conditions that would exist for a lunar mission, such as data rates, interactions with the robot, etc.

3 Configuration

One of the key elements to the success of a robotic mission is the configuration of the rover*. Most of the rover's performance metrics, such as maximum speed, terrain traversal ability, size, mass, etc., are determined by its

* Within the context of this report, a rover configuration is taken to mean the number and the relative position and orientation of the locomotion devices with respect to the body of the rover. For example, a solid rectangular body with four parallel wheels, two on each side, separated by a fixed distance, would describe the configuration of an automobile. Note that configuration is size independent.

configuration. However, as important as it is, no systematic means for selecting an appropriate rover configuration exists.

A typical methodology for determining an appropriate rover configuration uses a matrix showing relative strengths and weaknesses of the different configurations being considered. These tables present a list of significant criteria, a weighting of the criteria's significance and a score for how well each configuration satisfies the particular criteria. Although seemingly scientific, since hard metrics do not exist for many of the significant criteria, this method typically does little more than reflect the prejudices of the rover designer.

Another difficulty with matrix evaluations of rover configurations is the selection of criteria to be considered. Typical lists of criteria include, payload volume, stability, complexity, etc. There are three problems with this approach: First, a listing of unrelated criteria may fail to address the real issue, the ability of the particular configuration to carry out a specified mission. Second, inferring to a general principal from a list is difficult, and without knowing the general principal underlying the decision, the optimal configuration may not be chosen. Third, since the selection of an appropriate configuration is strongly mission dependent, lists of criteria developed for other missions may not be applicable.

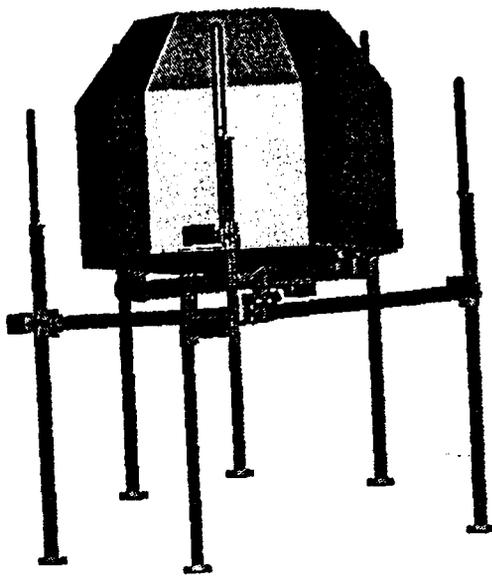
For the reasons listed, a matrix evaluation approach is not used to determine Daedalus' configuration. Instead, this document will focus on a particular configuration, the one developed to meet the specific needs of the mission. Although Daedalus is being built to carry out an Earth-based mission, the criteria for the lunar mission have a large impact on the system design and are also presented. The evaluation presented examines the proposed configuration in three broad categories: its ability to be delivered to the target area, its long term reliability and its capability for carrying out a mission. Section 3.1 introduces the Daedalus configuration and Section 3.2 is an evaluation of its capabilities. Sections 3.3 and 3.4 present the reasoning for selecting the lander/rover concept and a legged configuration respectively.

3.1 Daedalus configuration

Daedalus is pictured Figure 3.3-1. Daedalus belongs to a class of walking robots called frame walkers. This class of robots are typically considered to be the simplest walking machines capable of negotiating rugged terrain [Bar91]. The following discussion highlights some of Daedalus' key features in light of the three criteria mentioned above, its ability to be delivered to the target area, its long term reliability and its capability for carrying out a mission

Delivery:

- One of the key innovations of the Daedalus design is the concept of the integrated lander/rover. This delivery methodology combines traditional lander



and rover functionality into a single, integrated system. The benefits derived from this scheme include the use of smaller launch vehicle (reduced cost) for equivalent delivered payloads, simplified system integration and increased utilization of the landed mass. A number of technical issues required to actualize this concept are being concurrently studied.

- Daedalus' shape maximizes utilization of the launch-vehicle payload-fairing volume. To achieve compact stowage, Daedalus deploys a single leg after landing. To deploy this member, the horizontal drive motor in its typical operating mode, requiring no additional mechanisms. The Daedalus configuration scales over a wide range of launch vehicles. As a Daedalus-like rover gets larger, the mass fraction available for scientific payload increases.
- Daedalus, like many typical space craft, is axisymmetric to optimize utilization of the payload fairing and to simple spacecraft integration.
- The locomotion elements of the Daedalus concept consume less than 25% of the mass delivered by the launch vehicle and only a very small fraction of the payload volume.

Reliability:

- For a planetary mission, simplicity is an essential element of the rover configuration. For a mechanical system, it is generally true that the simpler the mechanism, the more reliable it will be. Although there are no hard-and-fast rules, simplicity is generally equated with mechanism degrees of freedom (DOF). Since a more complex rover mechanism is typically capable of negotiating more rugged terrain, there exists a simplicity versus capability trade-off.

- The Daedalus configuration is inherently statically stable, the center of gravity remains within the polygon formed by the feet in contact with the ground at all times.
- The mechanical components used are those that can withstand the harsh, lunar environment with little or no modification. Components requiring extensive re-engineering to withstand the lunar environment will typically be less reliable.
- Reliable mechanical systems utilize redundant components to reduce the likelihood of non-recoverable, system failures and minimize the number of single point failure modes. Daedalus uses redundant motors on all axes, allowing it to keep operating, albeit with diminished capabilities, in the event of a single motor failure.
- The Daedalus never has more than three legs supporting the body at any given time. This minimizes the potential for actuator conflict, which can be harmful to the system and consumes excess power.

Capability:

- The Daedalus configuration uses an orthogonal leg. The primary advantages derived from the orthogonal leg are that it allows decoupling horizontal and vertical motions, thus increasing power efficiency; and it eliminates shank rocking, which allows walking in rugged terrain.
- The only significant drawback to the Daedalus configuration is that the feet can not be independently placed. However, experience with this class of rover shows that this limitation is not as significant as expected. In the course of testing, the Martin Marietta frame-walker has yet to be in a situation where the fixed foot pattern has limited the its ability to continue making forward progress. The frame walker has demonstrated its ability of traversing almost any vector through Martin Marietta's Martian environment testbed in Denver, CO [CPS89].

3.2 Daedalus capabilities

This section describes Daedalus's walking cycle and kinematic capabilities. Daedalus has a design mass of 200 kg. This mass is divided equally among the four major subsystems: locomotion, power, computing and sensing, and scientific instrumentation. Daedalus stands between 1.5 - 2.5 m tall and is designed for a nominal walking speed of 10 m/min.

Figure 3.2-1 shows a schematic representation of Daedalus. The dark legs are those attached to the y-frame and the lighter legs are those attached to the body. The horizontal actuator has a stroke of approximately 0.9L (and not L, due to the volume consumed by the mechanism). The vertical actuator has a stroke of approximately 0.93L (and not L, again due to the volume consumed by the mechanism). For the Earth-based mission, L is approximately 1.25 m.

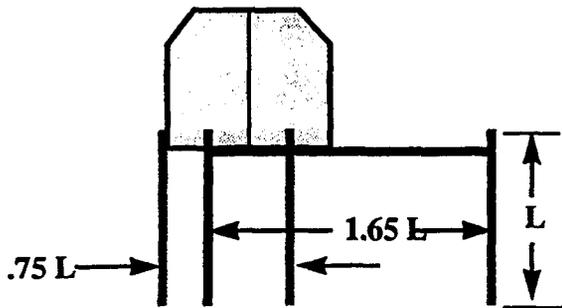


Figure 3.2-1 Schematic drawing of the Daedalus

3.2.1 Walking cycle

A complete cycle of motion for the walking beam is comprised of six distinct stages, where the transitions between the stages are called phases, with phase 1 being the transition between stage 1 and stage 2. Figure 3.2-2 presents two views of the walking cycle, a side view and a top view. In the top view, the legs contacting the ground are shown as solid circles and the legs in the air are shown as open circles. Although motion in one direction is shown, Daedalus can walk with equal facility in either direction. Daedalus moves forward $1.65 L$ per complete motion cycle.

3.2.2 Slope traversal

One of the design trade-offs for a walking beam is the length of the y-frame. For a fixed leg length, a longer frame enables longer strokes, but also limits slope traversal. For Daedalus, the length of the y-frame is dictated by stowage considerations. Maximum slope traversal is defined to be the steepest slope the rover can traverse while maintaining full stride with the body remaining level (i.e., the horizontal stroke is perpendicular to the local gravity vector). Figure 3.2-3 shows Daedalus traversing the steepest slope possible. Figure 3.2-3 and the following discussion require that the terrain be capable of supporting the applied loads. Thus, this discussion shows the limitations of the rover's motion independent of the terrain.

Two interesting facts come to light in Figure 3.2-3. First, maximum slope traversal is determined from the body fixed legs, not the y-frame fixed legs. The reason is

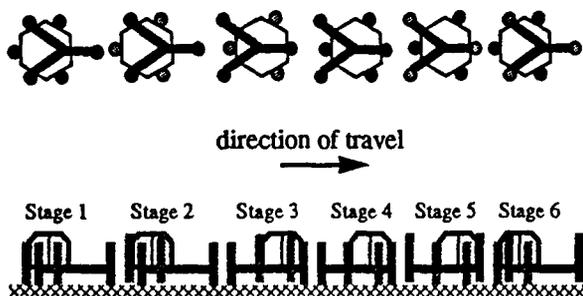


Figure 3.2-2 Walking Sequence

that the vertical stroke of the forward (in the direction of motion) leg is not fully utilized. Second, either set of legs can be used to provide the vertical motion, assuming that the horizontal impact of the feet with the terrain can provide a stable foothold.

Using the dimensions from Figure 3.2-1, the greatest longitudinal slope Daedalus can traverse is approximately 25 degrees. (Note, the Earth-based Daedalus is designed with slightly longer legs than those shown in Figure 3.2-1. It will be capable of traversing slopes of approximately 30 degrees.)

The greatest transverse slope traversable is a function of the leg length and the body width. For Daedalus, the greatest transverse slope traversable is in excess of 40 degrees. [Note, the actual Daedalus is designed with slightly longer legs than those shown in Figure 3.2-1 and transverse slope traversal will be in excess of 45 degrees.] Daedalus is capable of traversing its maximum longitudinal and transverse slopes simultaneously.

3.2.3 Step and ditch crossing

Similar analyses can be used to determine the tallest negotiable step and the widest negotiable ditch. Daedalus can negotiate steps of $0.93L$ in height if two conditions are met: the terrain at the edge of the step is solid and able to support the loads placed upon it and that there is a ledge behind the step that is approximately $1.75L$ deep.

A ditch is defined as a cut in the ground deeper than Daedalus's are long. The widest ditch that can be crossed is $0.5L$ wide. To perform this maneuver, several shortened steps are required. Like step climbing, ditch crossing also requires that the material along the edge of the ditch can support the applied loads

3.3 Lander/Rover

The traditional means for delivering a rover to a planetary body is to deliver the rover to the planetary surface as the payload of a dedicated landing stage. Examples of this include the Soviet Lunakhod, the Lunar Excursion Module of the Apollo program and the proposed Artemis common lunar lander. The biggest problem with these

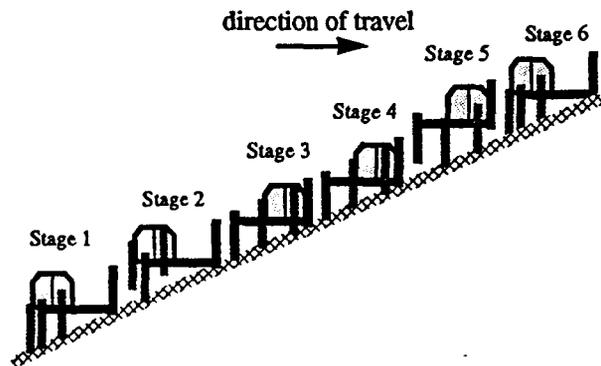


Figure 3.2-3 Walking Sequence - Steepest Slope Traversal

systems is the inefficient use of mass delivered to the planetary surface. Defining the delivered mass fraction as the delivered payload mass divided by the landed mass, and using the rover mass as the delivered payload mass, Artemis' delivered mass fraction is only 15 percent. (NASA plans to increase this to 45 percent for later missions.) The delivered mass fraction for the proposed lander/rover concept, however, is in excess of 85 percent. This means that the same payload can be delivered with smaller, less expensive launch vehicles, an important consideration given the current economic/political climate.

From the above discussion, it is clear that the greatest advantages of the lander/rover concept are accrued in the area of system delivery. The lander/rover concept also possesses certain advantages in terms of reliability. However, the lander/rover does not present any clear advantages in terms of the ability to carry out a mission other than reducing mission costs or allowing greater capability for the same cost.

Delivery

- The lander/rover concept allows larger payloads to be delivered with a fixed launch vehicle since a separate lander is not required.
- The lander/rover is simpler to integrate into a launch vehicle.

Reliability

- Shared components, such as computing, power, telemetry and structure eliminates unnecessary duplication of mission critical hardware. By reducing the component count, system reliability is typically increased. Using redundant components does not contradict the principal of shared components.
- The combined lander/rover eliminates the need to deploy the rover from the landing stage. Although the deployment maneuver does not represent a high risk, it is a single point of failure. With the combined lander/rover, failure to jettison the landing rockets and associated hardware will not kill the mission.

3.4 Legged rover

As stated earlier the configuration of the rover is a key element in determining mission success. Most of the rover's performance metrics, such as maximum speed, terrain traversal ability, size, mass, etc., are determined by its configuration. The design of an appropriate configuration for the APEX mission must satisfy two goals: the configuration must be capable of carrying out the prescribed mission and it must be amenable with the lander/rover concept.

Three means of locomotion are typically considered for robotic systems: legs, wheels and tracks. From Section 3.1, it is obvious that Daedalus is a legged rover. This section will highlight some of the reasons for choosing a legged system. Since tracks can be considered to be

large wheels, statements that apply to wheels also apply to tracked vehicles.

Delivery

- Legs are better suited to the lander/rover concept than wheels. The key concept behind the lander rover is the sharing of components. Although it is possible to envision landing on wheels that are suitable for planetary exploration, this would be a challenging problem, both in terms of materials and controls. A more likely solution would be to have a separate lander if a wheeled vehicle were used.
- Legs can be used to actively decelerate the robot as it touches down on a planetary surface. Although a challenging control problem, this deceleration by the legs simplifies the requirements for the descent rocket and descent control system; a problem that may prove to be more challenging than the leg control problem.
- Legs typically provide greater ground clearance than wheeled robots. This clearance is needed for rocket exhaust and a deceleration zone to prevent the body from hitting the terrain upon landing.
- The (typically) wider spacing of robot legs than robot wheels provides a more stable base for landing. The wider spacing is important to minimize tip-over on landing for the lander/rover concept.

Reliability

- Walkers suspend themselves over the terrain on discrete contact points, thus allowing them to avoid marginal footholds, to minimize dynamic shocks to the body, to maintain a stable pose independent of the under lying terrain and to simplify the perception and planing problems. Wheeled robots maintain continuous contact with the ground and must traverse all points along a line of travel, thus none of the listed benefits are achievable. These arguments assume a vehicle without an active suspension, a mechanism that would greatly increase the complexity and mass of a wheeled rover, but would allow it to realize some of these benefits. See [Bar91]
- The risk of immobilizing for a legged vehicle in soft terrain is significantly less than a wheeled vehicle. Should soft materials be encountered, such as sand or snow, it is more probable that a legged rover could extricate itself than a wheeled machine. [Bek56].
- Legs can be used to probe potential ground contact point before committing to them. By testing a foot-fall before using it, the probability of tip-over through ground failure is reduced.

Capability

- Exploration rovers should move continuously and "quickly". Although walking rovers are "slower" than wheeled robots on hard, flat surfaces, the speed advantage disappears in rugged and soft ter-

rains. For a certain range of speeds, the limiting factor in a walking machine's ability to traverse rugged terrain is the computing required to determine foot placement, since many of the surface irregularities do not alter the robots motion. For many wheeled machines, this is not true, because even in negotiating surmountable surface irregularities, the body is subjected to additional dynamic forces, forces that limit the robots ability to move faster.

- The ability of a legged vehicle to traverse rugged terrain is a function primarily of the rover's geometry. The ability of a wheeled vehicle to negotiate rugged terrain, however, is not only a function of its geometry, but also the underlying terrain. Thus, given a walking and rolling vehicle that can (kinematically) negotiate the same terrain, the walking robot will have greater terrain capabilities than the wheeled machine.
- Since the body of a walking machine is isolated from the underlying terrain, it can be precisely positioned. This is useful for gathering scientific data or pointing the antenna while moving.

4 Daedalus mechanical design

To validate the conceptual designs that arise from an evaluation of mission criteria, such as the requirements specified in Sections 2 and 3, the embodiment and detailed designs must be carried out. In theory there exists a very large, possibly innumerable, number of configurations that meet a set of design criteria, however, most of these configuration may be unrealizable. Before the Apex project committed itself to the Daedalus configuration, a number of the key design issues were explored in detail. A number of these are presented below.

4.1 Vertical and horizontal translational motions

The Daedalus configuration requires prismatic joints for its vertical and horizontal motions. A prismatic joint is comprised of a two basic parts: a moving element and a stationary element. The moving element is typically comprised of a strength member, bearing member and force transmission member and the stationary element is typically comprised of motor and bearing.

To reduce the total leg mass, Daedalus integrates the strength and bearing members into a single component. To determine the proper size for Daedalus' legs, equations describing the most probable failure mode are used. To use these equations, the leg material must be known a priori. Due to its strength-to-weight ratio, aluminum was chosen as the leg material, because for a given required strength, aluminum legs are typically lighter than steel legs. Since the legs are long, slender rods, under nominal operating conditions the most likely failure mode is buckling. Using appropriate safety factors, the required leg moment of inertia can be calculated.

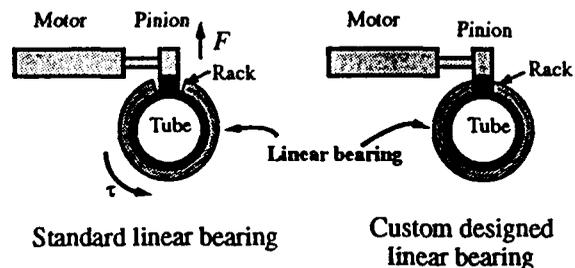


Figure 4.1-1. Linear Bearing Design

Ideally, Daedalus' legs remain vertical at all times, however, this may not always be the case. The legs must also be capable of supporting the applied bending load in those situations when the legs are not vertical. To determine a worst case angle, it is assumed that Daedalus is on the verge of tip-over. Again, using appropriate safety factors, the required leg moment of inertia can be calculated. Due to the relatively steep tip-over angle, the bending condition dominates the buckling condition.

There exist an infinite number of tubes that satisfy a given moment of inertia requirement. The desired solution is the one that is the lightest. Using this criterium, a tube with a very large diameter and very thin wall would be the best solution. However, such tubes are non-standard, are difficult to manufacture, are difficult to work with and are easily damaged. Instead, a standard size tube with a relatively thin wall was chosen.

To size the Y-frame, it is not the failure of the frame that is a primary concern, but rather its rigidity. To size the Y-frame, the maximum allowable deflection of the frame is determined from the kinematics of the robot. Using this information, a moment of inertia requirement for the Y-frame and horizontal tubes is calculated and then an appropriate tube selected.

To provide the force transmission, a gear rack is mounted directly to the leg. This design reduces the overall weight of the system, but also poses three challenges:

- Limiting the rotation of the leg to prevent transverse loading on the motor shaft
- Providing a hardened bearing surface
- Keeping the bearings lubricated and free of contaminants.

The use of traditional open, linear ball-bearing would result in a system that is subject to all of these difficulties. To alleviate the second and third problems, a solid linear bearings with Teflon based bearing surfaces is used as the anti-friction element. These bearings can be used on softer surfaces than ball bearings and are specifically designed for use in harsh environments where regular maintenance is not possible. The first problem is alleviated by using a custom bearing, see Figure 4.1-1, that prevents leg rotation.

4.2 Power train

To determine the sizes for the motor and gear train, certain assumptions about the robot's nominal walking cycle are needed, including its nominal speed and motion profile. For the APEX mission, a nominal speed of 10 m/min is desired and a trapezoidal velocity profile employed, Figure 4.2-1.

To minimize the maximum required locomotion power, the cycle time, determined from nominal speed, and the joint displacements, are used to determine appropriate phase times and acceleration times for each of the six phases. Other possible optimization criteria are possible, for example, minimum average power or minimum total energy, but minimum maximum power criterium was chosen because the typical space rated power sources are power limited, not energy limited.

The three primary components of power expenditure of a moving body are inertial power, gravitational power and frictional power. There are other sources of power expenditure, such as aerodynamic drag and rover/terrain interactions, but these affects are not considered. A non-regenerative system is assumed, thus the inertial and gravitational energies are not conserved. Using these assumption, the minimum maximum power expended to walk 10 m/min is found to be less than 75 W. It is important to note that this figure does not show the power expenditure for a body lift maneuver. Since lifting the body should only happen occasionally, the actuators are sized for the nominal walking cycle, and are geared to provide the higher torque required to lift the body. This will yield a slow body lift, assuming constant power, but this should have little impact on the overall mission.

To design the drive assembly components, knowledge of the power expenditure is not sufficient. Force and velocity profiles of the motions are also required. These

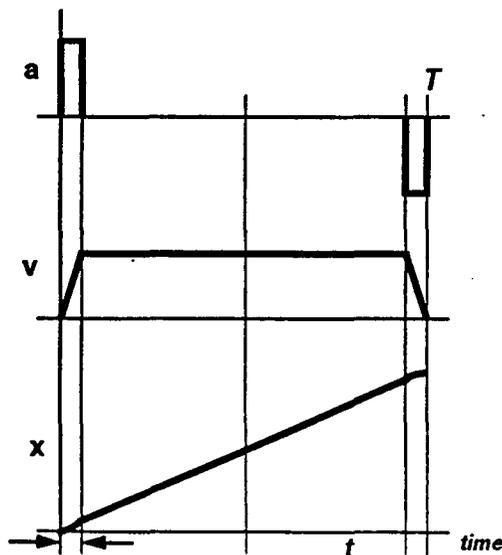


Figure 4.2-1 Kinematic Profile, Trapezoidal Velocity

profiles are readily calculated, and the drive components, motor, rack and gearbox, are designed accordingly.

For the Earth-based mission, brushed DC servo motors were chosen because they are simpler, provide a greater power to mass ratio and are less expensive than brushless motors. Although brush motors will not work for extended periods in a hard vacuum, there is no significant morphological difference between the two types of motors. Samarium-cobalt, aerospace motors were selected to achieve the required high reliability in the smallest possible motor.

Designing a single gearbox for Daedalus is a difficult problem to solve gracefully because of the two different motion profiles: high torque, low speed and low torque, high speed. One solution is to add a clutch that selectively engages different gearboxes for the different motion profiles. A combined gearbox/clutch system could be built to satisfy the requirements, but it would be fairly massive, would require extra electrical wiring and may not be very reliable. A better solution is to use a single epicyclic gearbox with two gear trains and two motors. This design has the advantage that the gearbox mass is reduced and the system is more reliable since the failure of either gear train does not disable the other and each axis has two actuators. Using this approach, a motor/gearbox package weighing less than 1.5 kg was developed that produces the required outputs. [Shi77]

4.3 Other design issues

4.3.1 Foot design

Proper foot design for a walking robot is of paramount importance. The foot must be capable of supporting the weight of the robot on a variety of terrains, must maintain good contact with the ground during the walking cycle and must house a variety of possible sensors. These issues are fairly complex and decisions about the foot can have a significant impact on other seemingly unrelated parts of the system. For instance, if walking requires selection of appropriate footfalls using the vision system, the diameter of the foot determines the minimum resolution of the vision system.

This paper will address only one of the issues regarding foot design, the shape of the sole of the foot. These results are a combination of some analyses and experience with a number of walking robots. The primary considerations for designing the sole of the foot are:

- Loading (contact pressure) - prevent excessive soil penetration, soil failure
- Traction (lateral forces) - sufficient for movement.
- Stability - prevent foot support failure

To determine the suitability of a given sole design, a spectrum of probable terrain types is required. This spectrum, however, is well represented by its extremes, soft terrain and hard terrain. Soft terrains will deform to shape of foot, and the shape of foot will impact traction,

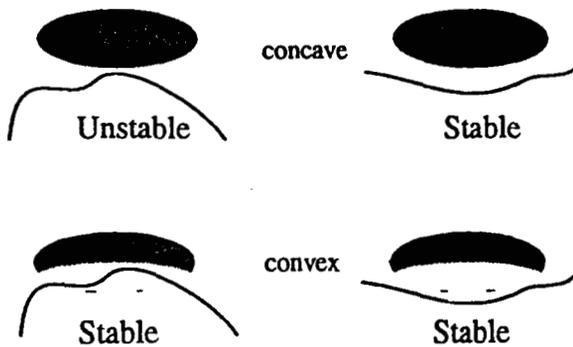


Figure 4.3-1 Stability of concave/convex sole on concave/convex terrain patches

however, traction is not expected to be a problem. For hard terrains, no rigid shape maximizes contact area for all cases.

There are two basic shapes that need to be considered for the sole, convex and concave. Figure 4.3-1 demonstrates that convex soles are stable over a wider range of terrain formations than concave soles. Non-rigid soles can provide excellent stability over a very wide range of terrains, however, the complexity of these designs makes them impractical for use on a planetary mission.

4.3.2 Electrical cabling

One of the goals of the vertical actuator design is to keep the number of wires to a minimum. By using brushed motors, instead of brushless, a multi-turn potentiometer, instead of an encoder, and preloaded micro-switch, instead of a force/torque sensor, the total number of required wires is significantly reduced.

To guide the wires from the moving part of the leg to the stationary base of the actuator, the cables must be managed in some fashion. This is a difficult problem to solve in an elegant manner. Currently researchers at NASA's Goddard Space Flight Center are working to develop electrically conductive gears, a clean solution for a problem such as this. Daedalus, however, will use plastic wire guides to manage the required cables.

4.3.3 Thermal design

One of the biggest challenges will be to keep the robot cool. Because of limited power and mass, the amount of heat that can be drawn off with active cooling systems is severely limited. It is necessary that the robot design minimize the amount of solar radiation impinging on the chassis. This can be accomplished in part with an "umbrella" of solar panels that moves so as to block the sun. Other measures taken are to cover the body with highly reflective materials and to employ power management techniques.

5 Summary

The design of a robot is a complex task. For any given scenario, there may exist a number of viable, potential candidate solutions. Based on previous experiences, the APEX project has decided to build Daedalus, a hexapod frame walker. This solution offers a number of advantages as compared to other possible solutions. The Daedalus design provides extraordinary capabilities for planetary exploration and for carrying out scientific agenda.

6 Acknowledgments

This work was performed, in part, under NASA Grant NAGW-1175. The authors would like to thank Mr. John Garvey of McDonnell Douglas Aerospace and Dr. Nicholas Colella of Lawrence Livermore National Laboratories for their support.

Bibliography

- [Bar91] J. Bares. *Orthogonal Walkers for Autonomous Exploration of Severe Terrain*. PhD thesis, Department of Civil Engineering, Carnegie Mellon University, May 1991.
- [Bek56] M. Bekker. *Theory of Land Locomotion*. University of Michigan Press, 1956.
- [CPS89] W. Chun, S. Price, and A. Spiessbach. Terrain interaction with the quarter-scale beam walker. In *SPIE Mobile Robots IV*, volume 1195, Philadelphia, PA, November 1989.
- [Man90] D. Manko. *A General Model of Legged Locomotion on Natural Terrain*. PhD thesis, Dept. of Civil Engineering, Carnegie Mellon University, April 1990.
- [MM88] Inc. Martin Marietta. Mars rover/sample return (MRSR): Mobility and surface rendezvous studies, October 1988. Work performed under JPL contract 958073.
- [Nag91] Peter V. Nagy. *An Investigation of Walker/Terrain Interaction*. PhD thesis, Carnegie Mellon University, August 1991.
- [Shi77] Joseph E. Shigley. *Mechanical Engineering Design*. McGraw Hill, 1977.
- [WL91] James R. Wertz and Wiley J. Larson, editors. *Space Mission Design and Analysis*. Space Technology Library. Kluwer Academic Publishers, 1991.

Drivetrain Design, Incorporating Redundancy, for an Autonomous Walking Robot

Gerald P. Roston¹
Kevin Dowling²

Abstract:

Carnegie Mellon University's Autonomous Planetary Exploration Program (APEX) is currently building the Daedalus robot; a system capable of performing extended autonomous planetary exploration missions. Extended autonomy is an important capability because the continued exploration of the moon, Mars and other solid bodies within the solar system will probably be carried out by autonomous robotic systems. There are a number of reasons for this - the most important of which are the high cost of placing a man in space, the high risk associated with human exploration and communication delays that make teleoperation infeasible.

A key component of a robot is its drive mechanism. This item, is critical in determining a robot's ability to efficiently explore its surroundings. The paper addresses the design of a redundant drivetrain for the Daedalus robot.

1 Introduction

Carnegie Mellon University's Autonomous Planetary Exploration Program (APEX) is currently building the Daedalus robot; a system capable of performing extended autonomous planetary exploration missions. Extended autonomy is an important capability because the initial exploration of the moon, Mars and other solid bodies within the solar system will probably be carried out by autonomous robotic systems. There are a number of reasons for this - the most important of which are the high cost of placing a man in space, the high risk associated with human exploration and communication delays that make teleoperation infeasible.

The Daedalus robot represents an evolutionary approach to robot design and incorporates key features from a number of predecessor systems, such as the CMU Ambler, the Martin Marietta frame-walker and others. Among other features, Daedalus combines the Ambler-derived orthogonal-leg design and the Martin walking-beam concept. Using technologies previously proven, on Earth, ensures that the required goals of reliability, terrainability and space relevance will be achieved.

In the course of developing Daedalus, a number of issues were highlighted and resolved. These issues include the ability to space-qualify the robotic system, to design

¹ Graduate student, Dept. of Mechanical Engineering/Field Robotics Center, Carnegie Mellon University, Pittsburgh, PA 15213

² Project Scientist, Field Robotics Center, Carnegie Mellon University, Pittsburgh, PA 15213

a power and mass efficient robot for carrying out scientific experiments, to economically deliver the robotic system on-board a commercial launch vehicle, to develop robust software capable of functioning for periods of weeks, to develop a system capable of stand-alone exploration missions and to enable planetary exploration by providing a general framework for autonomous mission planning.

This paper focuses on the actuator design, including motor selection and gearbox ratio selection. These components must be carefully chosen if the goals of extended autonomy are to be achieved. This paper first presents an overview of the proposed missions and the Daedalus robot, then describes, in detail, the actuator sub-system.

2 Mission Overview

The Daedalus configuration is designed to accommodate two missions, an extended duration lunar mission and a long duration Earth mission. The purpose of the Earth mission is to develop a system capable of performing long-duration, autonomous planetary exploration. Development for the Earth mission has commenced. Although the Daedalus robot itself is not space-qualified, only those component systems that are potentially space qualifiable are utilized.

Earth-based missions serve as analogs for lunar missions by simulating the operating conditions, terrains and interactions. Daedalus will be tested in the south-western US desert because the extreme ruggedness of the terrain is similar to that found on the moon. Candidate sites include Death Valley, CA; Kelso, CA and Cinder Lake, AZ. The goal of the earth mission is a multi-day, multi-kilometer, autonomous traverse of a region while performing selected scientific experiments. During this mission, every effort will be taken to simulate the actual conditions that would exist for a lunar mission, such as data rates, interactions with the robot, etc.

Lunar mission goals include the exploration of the lunar surface, performing lunar surface scientific experiments, site certification for follow-on missions and exploration of interesting formations such as volcanic vents, impact craters and lava tubes. Unless the lunar rover has the capability of storing large amounts of energy, or unless it possesses radioactive heat sources, the longest mission will last one lunar day (14 terran days) since the cold night temperature may damage certain system components. During the course of this mission, the rover is expected to cover upwards of 100 km over a variety of terrains.

3 Configuration

3.1 Daedalus configuration

Daedalus is pictured Figure 3.1-1 Daedalus belongs to a class of walking robots called frame walkers. This class of robots are typically considered to be the simplest walking machines capable of negotiating rugged terrain [Bar91].

3.2 Daedalus kinematic capabilities

Daedalus has a design mass of 200 kg. This mass is divided approximately equally among the four major subsystems: locomotion, power, computing and sensing, and scientific instrumentation. Daedalus stands between 1.5 - 2.5 m tall and is designed for a nominal walking speed of 5 m/min.

The greatest longitudinal slope Daedalus can traverse is approximately 30 degrees. The greatest transverse slope traversable is in excess of 40 degrees. Daedalus is capable of traversing its maximum longitudinal and transverse slopes simultaneously.

Daedalus can negotiate steps of greater than 1 m in height if two conditions are met: the terrain at the edge of the step is solid and able to support the loads placed upon it and that there is a ledge behind the step that is approximately 1.75 m deep.

The widest ditch than can be crossed is 0.6 m wide. To perform this maneuver, several shortened steps are required. Like step climbing, ditch crossing also requires that the material along the edge of the ditch can support the applied loads

4 Daedalus Actuator Subassembly

4.1 Vertical and horizontal translational motions

The Daedalus configuration requires prismatic joints for its vertical and horizontal motions. A prismatic joint is comprised of a two basic parts: a moving element and a stationary element. The moving element is typically comprised of a strength member, bearing member and force transmission member and the stationary element is typically comprised of motor and bearing.

To reduce the total leg mass and overall complexity, Daedalus integrates the strength and bearing members into a single component. To properly size these elements, equations describing the most probable failure mode are used. The gear rack, for power transmission, is bolted directly to the leg, and the leg assembly is driven by a gear motor with an output pinion.

4.2 Motion profile

To determine the sizes for the motor and gear train, certain assumptions about the robot's nominal walking cycle are needed, including its nominal speed and motion profile. For the APEX mission, a nominal speed of 5 m/min is desired and a trapezoidal velocity profile employed, Figure 4.2-1.

To minimize the maximum required locomotion power, the cycle time, determined from nominal speed, and the joint displacements, are used to determine appropriate phase times and acceleration times for each of the six phases of motion. Other possible optimization criteria include, minimum average power or minimum total energy. The minimum maximum power criterium was chosen because the typical space rated

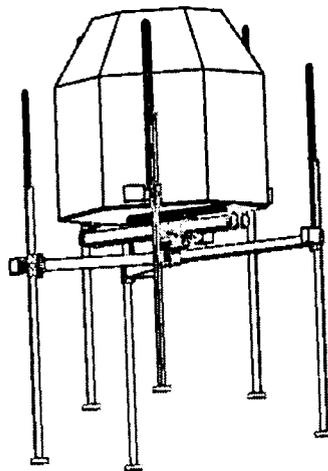


Figure 3.1-1 Daedalus

REV: 1, PLUMB-ACT

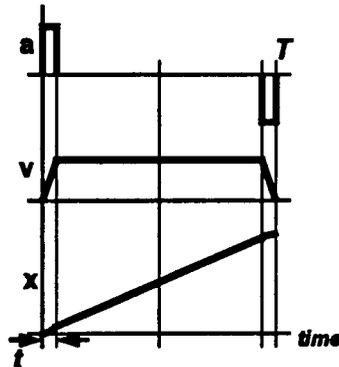


Figure 4.2-1 Kinematic Profile, Trapezoidal Velocity

power sources are power limited, not energy limited and to simplify the actuator motor/gearbox design.

The three primary components of power expenditure of a moving body are inertial power, gravitational power and frictional power. There are other sources of power expenditure, such as aerodynamic drag and rover/terrain interactions, but these affects are not considered. A non-regenerative system is assumed, thus the inertial and gravitational energies are not conserved. Using these assumption, the minimum maximum power expended to walk 5 m/min is found to be approximately 60 W, see Figure 4.2-2. It is important to note that this figure does not show the power expenditure for a body lift maneuver. Since lifting the body should only happen occasionally, the actuators are sized for the nominal walking cycle, and are geared to provide the higher torque required to lift the body. This will yield a slow body lift, assuming constant power, but this should have little impact on the overall mission.

4.3 Gearbox configuration

Because of the wide range of required force/speed, a two-speed gearbox will be used. This gearbox will provide two motion regimes, a high-speed/low-torque mode and a low-speed/high-torque mode. This is done so that the motors can be run at optimal efficiency in all motion phases.

The gearbox designed for Daedalus is a three stage planetary gear train. The first stage is a differential and is used to generate the difference in gear ratio that is required for the two modes of operation. The second two stages are speed reducers.

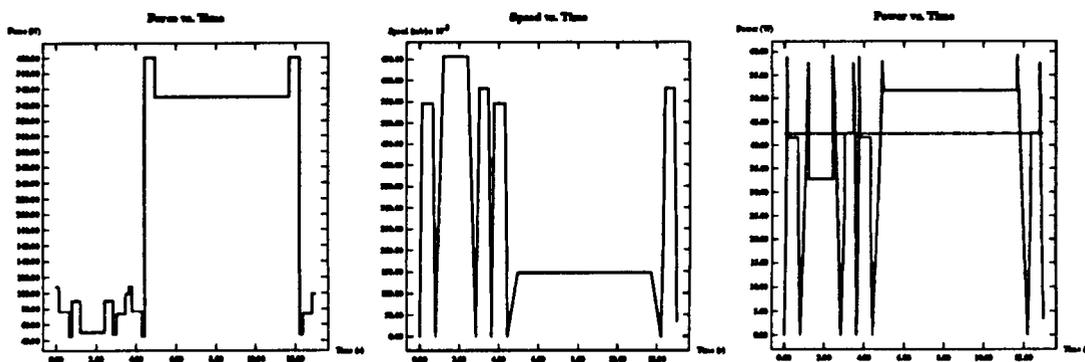


Figure 4.2-2 Speed, force and power profiles

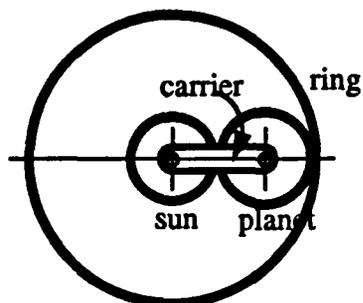


Figure 4.3-1 Planetary gear

Planetary gearing is a technique used to build gearing with high ratios, high torque carrying capability and good efficiencies into a compact package. The layout of a planetary gear stage is shown in Figure 4.3-1.

There are two ways to achieve the differential gearing for the first stage of the gearbox. The first is to use a clutch that selectively engages different gear ratios. The second is to use two motors and to selectively brake one or the other. The second approach was used because it provides redundancy in the case of the failure of a motor. Using this approach, a motor/gearbox package whose mass is less than 3.0 kg was developed that produces the required outputs. Figure 4.3-2 shows a schematic layout of the gearbox design.

Defining $\alpha = N_{ring}/N_{sun}$, the speed ratios (the speed of the input divided by the speed of the output) for the 6 combinations of inputs, outputs and fixed gears are shown in Table 1.1-1. (The torque ratios are simply the inverses of the speed ratios.):

Input Gear	Fixed gear	Output gear	Speed ratio
sun	carrier	ring	$-\alpha$
ring	carrier	sun	$-1/\alpha$
carrier	sun	ring	$\alpha/(1 + \alpha)$
ring	sun	carrier	$(\alpha + 1)/\alpha$
sun	ring	carrier	$1 + \alpha$
carrier	ring	sun	$1/1 + \alpha$

Table 4.3-1 Speed ratios

4.4 Motor specification

4.4.1 Motor requirements

The motors to be used should meet the following requirements:

1. They must be as small and light as possible
2. They should not require more than 48 VDC
3. They should not require more than 7 A during normal operation
4. Temperature rise during normal operating conditions should not exceed 35°C
5. The motor must not have an integral fan or require one for cooling
6. The cost of the motors and required amplifiers must be “reasonable”

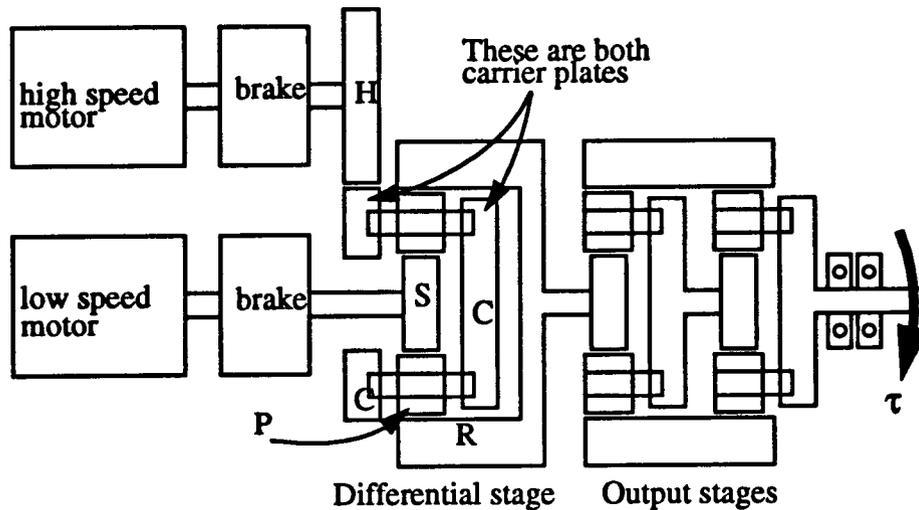


Figure 4.3-2 Gearbox schematic

4.4.2 Motor requirements

To develop the motor requirements, the non-accelerated portion of the motions will be considered. This is done because these periods dominate the walking cycle. Table 4.4-1 shows the required speed and force for each of the four types, not phases, of motion. These types are lifting/lowering the legs, moving the y-frame, lifting the body and moving the body. The first two types of motions (referred to as high speed motions) are assumed to use one of the gearing ratios and the second two types of motions (referred too as low-speed motions) use the other gearing ratio. For lifting/lowering the legs and moving the y-frame motions, there are two forces shown. The italicized force is that force which yields the same power expenditure as the body move for the specified speed. The speed for the body lift is chosen to yield the same power as a body move. Choosing pinion gears, 0.625 in radius for vertical motions and 0.375 in radius for the horizontal motions, allows rewriting the requirements in Table 4.4-1 as torques and angular velocities.

	high speed motions	low speed motions
vertical motors	$f = 75$ (92) N [$\tau = 0.88$ Nm (124 oz in)] $v = 0.565$ m/s [$\omega = 59.3$ r/s (566 rpm)]	$f = 1000$ N [$\tau = 9.53$ Nm (1349 oz in)] $v = 0.052$ m/s [$\omega = 5.5$ r/s (52 rpm)]
horizontal motors	$f = 50$ (79) N [$\tau = 1.25$ Nm (177 oz in)] $v = 0.656$ m/s [$\omega = 41.3$ r/s (395 rpm)]	$f = 350$ N [$\tau = 5.56$ Nm (787 oz in)] $v = 0.148$ m/s [$\omega = 9.3$ r/s (89 rpm)]

Table 4.4-1 Force/torque and speed requirements

4.4.3 Optimal gear ratio selection

Motors have an operating point at which they are most efficient. The purpose of gearing is to change the required output conditions to the motors optimal operating conditions, if possible. For this robot, however, it is not possible to have optimal performance for all loading conditions because of the disparity between the high speed and low speed motions. However, by determining the optimal gearing ratios for three of the phases, the overall system performance can be optimized.

For a first order approximation, a motor can be defined by four parameters: torque sensitivity (k_T), static friction torque (τ_f), viscous damping (τ_d) and coil resistance (Ω). Back EMF (k_e) is the same as torque sensitivity. Equation 4.4-1 shows the current required for a given load torque (τ) at a given angular velocity (ω) with a gear ratio of (N), the required voltage and the motor's percent efficiency.

$$i = (\frac{\tau}{N} + \tau_f + \tau_d \omega N) / K_T \quad V = i\Omega + K_e \omega N \quad e = \frac{\tau \omega}{Vi} \quad 4.4-1$$

To determine the gear ratio with the highest efficiency, calculate

$$\frac{\partial e}{\partial N} = 0 = (2k_T^2 \tau_d \omega^2 + 2\Omega \tau_d^2 \omega^2) N^4 + (k_T^2 \tau_f \omega + 2\Omega \tau_d \tau_f \omega) N^3 - 2\Omega \tau_f N - 2\Omega \tau^2 \quad 4.4-2$$

and solve for N . Since this is a fourth order equation, solving for the roots symbolically is difficult, so a numerical approach should be used.

Since the motion that consumes the most power is the body move, the load conditions for that motion are the ones to be used in Equation 4.4-2 to determine the low speed gear ratio. It must also be recognized that the gear ratio developed is a theoretical ratio in the sense that it may not be achievable given the constraints on gear design from the manufacturer. However, small changes in the ratio will not have an appreciable impact on efficiency.

To determine the high speed gear ratio, Equation 4.4-2 is used with both high speed motion requirements. The two resulting gear ratios are averaged to yield the high speed ratio. The low speed ratio divided by the high speed ratio yields the difference in ratios. This approach maximizes the efficiencies of the motions used for walking at the cost of the body lift motion.

4.4.4 Actual implementations

The following sections show the results of these calculations applied to an actual motor. The required variables are determined from the motor data sheets, then the theoretical optimal gearing ratios are determined, the ratio between the gear ratios is selected found. These values are used to determine motor efficiencies for the four types of motions. As a final step, an actual gear ratios (based on manufacturers' constraints) are used to determine actual system performance.

The motor selected is a Pittman 4111 with winding 2. This motor is a brushless DC servo motor. It is a square motor, 40 mm x 40 mm x 67.8 mm and has a mass of 380 gr. The optimal difference between the ratios is found to be 6.31:1. The parameters for this motor are $k_T = 0.0314 \text{Nm}$, $\tau_f = 0.0013 \text{Nm}$, $\tau_d = 2.6 \times 10^{-6} \text{Nm}$ and $\Omega = 1.21 \text{Ohms}$. The temperature rise per watt is 4.1°C . The calculations used in Table 4.4-2 show that the motor operates between 25 and 35 VDC and draws between 1.5 and 3 Amps.

In theory, any combination of gear sizes can be used to construct a planetary gear. The particular manufacturer for the Daedalus gearbox, CGI Incorporated, uses a small number of sun gears (12, 18, 24, 30, 36, 42, 48 and 54 teeth) matched with a single ring gear (108 teeth) to produce a wide variety of possible gear ratios. To control costs, only standard gear ratios were considered for the Daedalus project. This resulted in the following selection of gearing: the differential stage sun, 30 teeth; the output stage sun gears, 24 teeth; gear H, 129 teeth and gear C, 95 teeth. This yields actual overall gear

ratios of 108.90:1 for the low speed motions and 17.43:1 for the high speed motions.

variable	units	Load condition			
		horizontal, low speed	horizontal, high speed	vertical, low speed	vertical, high speed
Gearing ratio (theoretical)	N	108.400	17.179	108.400	17.179
Load torque (theoretical)	N m	5.560000	1.250000	9.530000	0.880000
Load speed	r/s	9.300000	41.300000	5.500000	59.300000
(Motor torque)	oz in	7.262878	10.303229	12.448782	7.253473
(Motor speed)	rpm	9626.5378	6774.9776	5693.1138	9727.7524
Efficiency	%	87.05	84.73	81.61	87.06
Gearing ratio (actual)	N	108.90	17.431	108.90	17.431
Load torque (actual)	N m	5.560000	0.793000	9.530000	0.714000
Load speed	r/s	9.300000	41.300000	5.500000	59.300000
(Motor torque)	oz in	7.262878	6.536368	12.448782	5.885204
(Motor speed)	rpm	9671.2411	6874.5415	5719.5512	9870.7097
Efficiency	%	87.04	86.32	81.71	86.51
Temperature rise	C	31.55	21.28	48.11	27.07

Table 4.4-2 Motor efficiency for the four motion types

The calculated temperature rises assume 100% duty cycle. When the actual duty cycles are taken into consideration, this motor yields acceptable results given the initial criteria, Section 4.4.1. The cycle time weighted efficiency of the motor is 86.8%. In addition, the least efficient motion is the vertical body lift, and this is perfectly acceptable as this motion occurs the least frequently.

4.5 Brake sizing

Although back-driving the brakes from the output shaft is highly unlikely, the possibility of back-driving one brake through the differential stage of the gear box while the other motor is operational presents a likely scenario. Consider first the case when the high speed motor is operational. This occurs when the legs are being lifted/lowered or the y-frame is being moved. From Table 4.4-1, the latter case requires greater output torque, 1.43 Nm. From Equation 2.2-2, the required brake torque is given by

$$\tau_{brake} > \tau_s = \frac{1}{\alpha} \tau_r = \frac{1}{\alpha} \frac{\tau_{load}}{N} = 0.014 \text{ Nm}, \quad 4.5-1$$

where α is defined in Table 4.3-1 and N is the gear ratio, Section 4.4.4.

Now consider the case where the low speed motor is operational. This occurs when the body is being lifted/lowered or the body is being moved. From other documentation, the former case requires greater output torque, 9.53 Nm. (Note: this value is a steady state value and does not include accelerating the body. However, body lifts are very slow and the additional force due to the acceleration is less than 4% of the steady state force.) Using Equation 2.2-3, and incorporating the additional spur gear

pair, $N_1 = 129/95$, the required brake torque is given by

$$\tau_{brake} > \tau_c = \frac{1+\alpha}{\alpha} N_1 \tau_r = \frac{1+\alpha}{\alpha} N_1 \frac{\tau_{load}}{N} = 0.591 \text{ Nm} . \quad 4.5-2$$

Using the same equation, the brake torque required to move the body is 0.392 Nm.

Two options were considered for the selection of the brakes. The first was to use an Electroid bi-stable brake, model BSB-3, which can provide 0.28 nm of holding torque. This brake is unique in that it does not require constant excitation to remain released. To operate the brake, an electrical pulse, 24 VDC at 2.1 A for 100 ms is applied. For a typical motion, the brake will be released then re-engaged, resulting in a total energy expenditure of 10 J.

Another option is to use a fail-safe brake, a Binder 86 621 A04. This brake requires 8 W of power to release. Comparison of the two brakes shows that if a motion last for more than 1.25 seconds, using the bi-stable brake will result in a less energy consumption. Using the optimal cycle times, using a combination of fail-safe and bi-stable brakes yields the most energy efficient operation, 140 J per cycle, where using fail-safe brakes only requires 154 J per cycle. However, the simpler electronics for the fail-safe brakes makes them a more attractive solution.

5 Summary

Careful design of a robot drivetrain is essential for optimal performance. To achieve the goals of a planetary exploration mission, the robot drivetrain must operate efficiently, robustly and incorporate redundancy. The design procedure outlined was used to develop a drivetrain, using mostly shelf available components, that results in a system efficiency greater than 65%. These techniques are generic and can be applied to the drivetrain of other systems.

6 Acknowledgments

This work was performed, in part, under NASA Grant NAGW-1175. The authors would also like to thank Mr. John Garvey of McDonnell Douglas Aerospace and Dr. Nicholas Colella of Lawrence Livermore National Laboratories for their support.

Bibliography

- [Bar91] J. Bares. *Orthogonal Walkers for Autonomous Exploration of Severe Terrain*. PhD thesis, Department of Civil Engineering, Carnegie Mellon University, May 1991.
- [Bek56] M. Bekker. *Theory of Land Locomotion*. University of Michigan Press, 1956.
- [CPS89] W. Chun, S. Price, and A. Spiessbach. Terrain interaction with the quarter-scale beam walker. In *SPIE Mobile Robots IV*, volume 1195, Philadelphia, PA, November 1989.
- [Man90] D. Manko. *A General Model of Legged Locomotion on Natural Terrain*. PhD thesis, Dept. of Civil Engineering, Carnegie Mellon University, April 1990.
- [WL91] James R. Wertz and Wiley J. Larson, editors. *Space Mission Design and Analysis*. Space Technology Library. Kluwer Academic Publishers, 1991.

0 Introduction	1
1 Mission Overview	2
1.1 Lunar Mission	2
1.1.1 Delivery requirements	2
1.1.2 Lunar mission miscellany	3
1.2 Daedalus Mission	3
1.2.1 Site Selection	4
1.2.2 Delivery requirements	4
1.2.3 Surface Mission Miscellany	5
1.3 Feature Concept	5
1.4 Mission Scenario	6
2 Configuration	8
2.1 Lander/Rover	8
2.2 Legged rover	9
2.3 Daedalus configuration	10
2.3.1 Comparison	13
2.3.2 Design considerations	15
2.3.3 Power, stability and ground-clearance	16
2.4 Daedalus capabilities	17
2.4.1 Walking cycle	17
2.4.2 Slope traversal	17
2.4.3 Step and ditch crossing	18
2.5 Scout	20
2.5.1 Motivation	20
2.5.2 Possible Scout Designs	23
2.5.3 Conclusions	26
3 Daedalus mechanical design	29
3.1 Vertical actuator	29
3.1.1 Assembly requirements	29
3.1.2 Component design	34
3.1.3 Component manufacturing	37
3.1.4 Sensors	39
3.1.5 Electrical cabling	39
3.1.6 Control	40
3.2 Horizontal actuator	40
3.2.1 Requirements	40
3.2.2 Component design	41
3.2.3 Component manufacturing	41
3.2.4 Sensors	41
3.2.5 Electrical cabling	41
3.2.6 Control	41
3.3 Body and sensor mast	41
3.4 Foot Design	41
3.4.1 Introduction	41
3.4.2 Past Experience	41
3.4.3 Foot Sensing	43
3.4.4 Foot Size	45
3.4.5 Foot Shape	46
3.4.6 Foot Shape Study	46
3.5 Thermal design	48
3.5.1 Introduction	48

3.5.2 Design Objectives	48
3.5.3 Design Consideration	49
3.5.4 Heat Sources and Consideration on Design Parameters	49
3.5.5 Thermal Control Techniques	52
3.5.6 Designs	56
3.5.7 Examples of NASA experiences with Heat Pipe Systems	62
3.5.8 Concluding Remarks	62
3.5.9 References	62
4 Navigation Sensors	64
4.1 Design Rationale	64
4.2 Sensing Tasks	64
4.3 Comparison of Map Building Sensors	65
4.3.1 Range Sensors for Map Building	66
4.3.2 Configuration	67
4.4 Elevation Map Generation	68
4.4.1 Multi-Baseline Stereo	68
4.4.2 Converting Depth Maps to Elevation Maps	71
4.4.3 Map Maintenance	72
4.4.4 Calibration	73
4.5 Comparison of Position Estimation Techniques	73
4.5.1 Global position estimation	73
4.5.2 Local position estimation	78
4.6 Teleoperation and mechanism survey	79
5 Planning and Navigation	80
5.1 Overview	80
5.2 Global Route Planning	80
5.2.1 Maps	80
5.2.2 Global Positioning	81
5.2.3 Global Navigation	82
5.3 Local route planning	86
5.3.1 Specifications	87
5.3.2 General Philosophy	87
5.3.3 Resolution	87
5.3.4 Input	88
5.3.5 Output	89
5.3.6 Algorithms	89
5.3.7 Error Recovery	89
5.3.8 Dense Obstacle Fields	90
5.3.9 Simulation	90
5.3.10 Conclusions	90
5.4 Positioning on the lunar surface	91
5.4.1 Star Tracking	91
6 Computing and Electrical Hardware	93
6.1 Control System	93
6.1.1 Sensors	93
6.1.2 Communications/Telemetry	93
6.1.3 Computing	93
6.1.4 Actuators	94
7 Telemetry	96
7.1 Purpose of Telemetry and Tele-operation	96
7.2 Fitting Telemetry into the Architecture	96

7.3 Mission Issues	97
7.4 Daedalus to Relay Station	98
7.4.1 Wireless Ethernet Link	98
7.4.2 Commercial Cellular System	99
7.4.3 Ku-Band Line of Sight Link	99
7.4.4 Radio/TV Link	100
7.5 Relay Station to Mission Control	101
7.6 Streams for Data and Control	102
7.6.1 Data Streams	102
7.6.2 Command Stream	103
8 Scientific Instrumentation	105
9 Power Requirements	106
9.1 Requirements	106
9.1.1 Mission	106
9.1.2 System	106
9.1.3 Power Generation	108
9.2 Energy Storage	109
9.3 Regulation and Control	109
9.4 Sample Mission	110
9.5 Conclusion	112
10 Software	113
10.1 Software design	113
10.2 Architectures	113
10.2.1 Task Control Architecture	113
10.3 Architecture modules	114
10.3.1 Image sensing manager	114
10.3.2 Controller	115
10.3.3 Base Station Manager	115
10.3.4 Planners	115
10.3.5 Mission manager	115
10.3.6 Data manager	115
10.4 Control	115
10.4.1 Physical (low-level) control	115
10.4.2 Reactive gait planning and motion control	123
10.5 Simulation	128
10.5.1 Environment Simulation	128
10.5.2 Mechanism Simulation	129
10.5.3 Software Module Simulation	130
10.5.4 User Interface	132

0 Introduction

This document is the end result of a semester long activity in the Mobile Robot Design graduate course. The schedule for the 100 day activity was to present a candidate design for the APEX (A Planetary Explorer) project, have a complete mission scenario, and support technical decisions surrounding the design. The work performed in this course is a direct contribution to ongoing planetary robotics research at CMU.

The Ambler, a six-legged walking machine was developed by CMU for demonstration of technologies required for planetary exploration. In its five years of life, the Ambler project brought major breakthroughs in various areas of robotic technology. Significant progress was made in:

- mechanism and control, by introducing a novel gait pattern (circulating gait) and use of orthogonal legs
- perception, by developing sophisticated algorithms for map building
- planning, by developing and implementing the Task Control Architecture to coordinate tasks and control complex system functions.

In September 1992, the Ambler walked 0.5 Km fully autonomously on outdoor rolling terrain. In that last walk we demonstrated the integration of multiple systems to provide robot autonomy.

The APEX project which is the successor of the Ambler project, sets more challenging goals for the planetary robotics group. Our objective is to develop a system that will demonstrate autonomous and reliable operation for long term missions.

1 Mission Overview

The Daedalus configuration is designed to handle two missions, an extended duration lunar mission and a long duration Earth mission. The lunar mission is planned for within five years. The system development for the Earth mission has already commenced. The purpose of the Earth mission is to develop a highly reliable, autonomous system for rugged terrain exploration. The systems and concepts developed for the Earth mission will be used for the lunar mission.

1.1 Lunar Mission

While this mission is still many years away, much of Daedalus is being designed and developed with a potential lunar mission in mind. Issues such as thermal regulation, positioning, mechanical structure, and telemetry all consider both missions. This process has the advantage of allowing the robot to use the results of the earth mission to prove the potential for success in the lunar mission.

The goal of the lunar mission is to provide a clearer understanding of some of the geological features and history of the moon itself. Since a manned mission is too costly and expensive for these times, and the only way to gather information is to actually wander about on the surface, a robot is the logical choice.

While exploring the geological features such as vents, craters, rills, and rubble, it is also being considered that the robot will look for lava tubes on the lunar surface. While these features are not ensured to exist, the discovery of a vacuated tube would provide a safe and reasonable place for a haven. This haven could be used by the robot to survive the cold of the lunar night, and would also provide a haven for astronauts seeking shelter from galactic energies while building a manned base on the lunar surface.

1.1.1 Delivery requirements

- Combination Lander/Rover

The lunar delivery of the Daedalus robot is unique in that the rover and the lander are combined on one platform. This allows a smaller total payload and, as a result, a smaller launch vehicle and a cheaper pricetag. The feasibility of using the robot as a lander is being studied as a separate project. The main considerations are the ability to ensure a feet first landing, the ability to slow the descent speed to a value within the robot leg stress limitations, and the ability to jettison the thrusters and the fuel tanks upon landing.

- Payload shock

During the stages of separation of the launch vehicle boosters, and during the lift-off and the landings, the robot will undergo high G-force loads. Thus, all hardware on the robot must be designed and built to withstand these shocks. Protecting the cameras and the solar panels as well as items like disk drives and other scientific instrumentation is a major issue.

- Payload size

Due to the combination of a rover/lander, the majority of the payload becomes functional. The shape of the robot and the retractability of all legs and arms is optimized to provide the largest possible container within a standard rocket cargo area. Most of the robots mechanism is outside of the central housing, which can hold a sizable payload of scientific equipment as well as the hardware for the robot controller.

- Timing of Lift-off

To achieve a long duration mission, the robot needs to land on the moon close to sunrise on a lunar day. This restriction can be projected backwards to derive the suitable dates for a launch using standard orbital mechanics.

1.1.2 Lunar mission miscellany

The goal of the lunar mission is to explore new sections of the lunar surface, and to perform lunar science which cannot be performed without visiting the surface. Additionally, a subgoal is the location and identification of a lunar lava tube which might serve as a shelter or base site for a permanent manned station.

- Duration = One Lunar Day

Since power and temperature are prime concerns, and during the lunar day the sunlight provides an unending power and heat source, the mission is designed to last for one lunar day which is equivalent to fourteen earth-days. This equates to 360 hours of continual operation by the robot. During the ensuing night time, it is hoped that the robot might find a shelter from the cold and be able to restart again fourteen days later. However, this would be a benefit which is unplanned for in the current mission concept.

- Power Supply

The main source of power would be provided by an extending solar umbrella. The backup power source will be a rechargeable battery. During the times when the robot is standing still, either planning or waiting for a teleoperated command, solar power can recharge the batteries. In this manner, the robot should be able to explore and carry out mission agenda for a long period of time.

- Thermal Regulation

To eliminate the extra heat generated by the on-board hardware and computing and by the solar energies, radiation techniques will be used. In this method, a shaded side of the robot which is open toward space will have a near zero kelvin temperature. So convection cooling is established from the exposed and heated sections of the robot to this shaded or dark area.

- Communication

An antenna and on-board satellite dish are the least likely forms of communication hardware for the robot. The power and complexity needed to servo the dish and antenna in the proper direction would be too costly. The more likely approach is to deploy one of two base stations with such a servo dish at the landing site. The other base station should be deployed at some maxima height location in the observed terrain to allow greater range, etc. The power of the antenna and size of the dish will be determined by payload considerations and telemetry requirement.

- Global Position

skyline tracking, horizon feature extraction or sextant/star tracking algorithms performed either on board, or by a human via the teleoperated approach. This is perhaps the most difficult issue since both the resolution of lunar maps and methods of locating position from the surface are both much less precise than on earth.

1.2 Daedalus Mission

The Earth mission is planned for the summer or fall of 1994 at a site which has yet to be determined. The mission length should last about 18 days with 15 of those being actual operation of the robot in an autonomous situation. The mission has two objectives:

- Autonomous exploration over rugged terrain for an extended period of time.
- Location and tracking of several features valuable to geological science.

Proving the effectiveness of mobile robots in a harsh, rugged environment is a long standing goal of the Robotics Institute (RI) and the Field Robotics Center (FRC) at Carnegie Mellon University. In addition, demonstrating the value and flexibility of robots in assisting with scientific research and gathering information in sites inaccessible by humans is another focus of the program.

Previous efforts have proven that reliability can be obtained (Ambler) while walking in an outdoor environment, but never at speeds which would allow much exploration to be accomplished in a short moment of time. With a top walking speed of ten meters per minute and an operation time of about 15 hours per day, Daedalus should be able to cover eight kilometers of rugged terrain over a single day.

1.2.1 Site Selection

Several sites for the earth mission are being considered, including a lava tube field in New Mexico (El Ma Pais), a sand dune and volcanic site in California (Baker), the epitome of American wasteland/desert (Death Valley) along with several others. The issues being addressed in this decision are discussed below.

In order to run the robot, permission to perform the mission must be obtained. Many national parks or monuments prohibit humans wandering off the trails, etc since they might damage the beauty of the site. If private lands are potentially being traversed clearly this is a major issue.

Since the earth mission is a prelude to a lunar mission, the initial site should have the same basic appearance as the lunar surface. In this regard a minimal amount of vegetation should be present on the site. This is especially a concern due to the use of stereo imaging as a navigation device. Clearly the robot can walk through a flat plain covered in small shrubs, but to the perception modules this may appear to be an impassable rubble field.

Due to the inavailability of solar energy as a reliable and constant energy source on the earth, and the ability of earth weather to damage solar panels, the robot will be battery powered. In order to save weight, only about eight hours of power will be on board at a time. This means that to complete a fifteen hour exploration day, the batteries must be switched in the early afternoon. For this reason, the site which is chosen must have access for a rescue team to get to the robot. Presumably an off road vehicle will be used for this, and so driving over the site must be possible (or at least driving to several rescue points must be possible).

To demonstrate the ability of Daedalus to handle rugged terrain, the site must contain said rugged terrain. In addition, a variety of soil types ranging from lava to rock to sand to earth to rubble would give a good test to the system.

To perform feature tracking and scientific studies, there must also be something to study in the area. Thus the site needs to have rills, craters, dunes, lava tubes, vents, faults, exposed rock, or other items for the robot to discover in its exploration.

Another fundamental issue is that the terrain should be uninhabited by any creatures large enough to cause problems to the robot. This includes hikers or other humans who may be curious. The perception modules will have a hard time dealing with moving obstacles which don't appear in consistent places in the stereo images. In addition, the terrain should not have any fences or other obstructions which could easily be missed by the stereo image and cause severe problems for the robot.

Finally, the climate at the chosen site during the mission time must be one which is consistently hot, dry, and sunny. These features again mimic the lunar surface and provide the best test of the systems as well as providing an uninterrupted test sequence. Should the robot have to weather an evening thunderstorm and maneuver in dim lighting with high humidity and potential water droplets on camera lenses the results could be disastrous for the mission objectives.

1.2.2 Delivery requirements

- Transport and packaging

Not nearly as specialized as on the lunar mission, the robot needs to be lifted by several individuals and placed in a truck or other form of conveyance. Possibly it should be positioned on a cushioned bed and strapped down to prevent tipping. However, the robot should be able to take the standard shocks of driving on highways so no extra precautions need to be made.

- Costs

The transportation and deployment of the earth mission should result in little more than the minimal costs. Potentially a truck needs to be rented with gas and driver paid but the entire trip should cost less than a thousand dollars from CMU to anywhere in the southwest U.S.

1.2.3 Surface Mission Miscellany

- Weatherproofing

Since the robot will be operating in an atmospheric environment, care must be taken to ensure that humidity and dust as well as wind and rain do not impede the robots progress. The lenses of the cameras should be sheltered and the tracks for the leg mechanisms need to be covered to prevent terrain features from interfering and causing damage to the robot.

- Night Day Shifting, Sun angles

Over a week period, the robot will pass several day and night cycles. The orientation of the sun will also change much more rapidly than it would on the moon. These areas affect the use of solar cells as well as the ability to truly test the robot under continual operating conditions.

- Power Supply

The main source of earth power will be batteries. Since the delicate solar panels used on a lunar surface could easily be broken by harsh weather, or made useless by overcast skies, the only robust power source for the untethered robot is batteries. These will probably be switched daily and these battery switches would be the only human intervention to the robots normal routines.

- Thermal Regulation

It would be possible to use fan cooling, but to stay in line with the needs of the lunar mission, the robot will likely use a radiative method of cooling along with normal convection. The most probable method involves flat heat pipe.

- Communications

In addition to a satellite communication, wireless and radio communications are also possible. However again, to prove space readiness, there will be a satellite link to a ground station for proving that high time delays dont affect the control or the monitoring of the robot.

- Global Position

Using GPS would solve this problem, but without a lunar analog it is a hesitant technology. Other possible solutions are feature matching or teleoperated position information. The latter is probably more accurate and more guaranteed, however the development of a feature matching algorithm would provide for much greater system robustness.

1.3 Feature Concept

A simple method for determining the destination of Daedalus is described below. Note that this may not be the actual implementation of the robot, but the concept should be much the same. The goal of the feature concept is to have Daedalus always headed toward an item of interest for its scientific payload and objectives.

The basic premises for which the concept operates are fairly common-sensical. Should that item of interest be inaccessible or turn out to not be interesting it is left and another item becomes the new destination. However, if a very interesting item is noticed en route to something else of less

value, the new item should receive top priority. If the robot has run out of interesting features it should have several original locations marked as objectives on the global map toward which it can traverse.

Before the mission, a list of potential features and how to recognize them are included in the perception module. The global map of the mission site is marked with several interesting places to explore should the robot be in the area. The list of features should be ranked and weighted to allow the robot to decide amongst conflicting features (human intervention could also be used to force Daedalus to go to a desired location).

The ranking of features and the actual list of features depends largely on the scientific payload which is carried. However, even without any payload at all there is much which could be done on the mission. The sensors used for navigation and control of the robot are also able to provide information regarding the terrain and structure of the path it follows.

A sample list of terrain features ranked from high potential for scientific use toward lower potential for a basic earth mission are listed below:

- exposed strata or vein providing geologic history
- cave or opening providing possible shelter
- thermal vent providing information on internal structure
- (on moon, an upwelling from impact crater providing meteorite information)
- rills or canyons or sinkholes providing possible paths to higher ranked features
- boulder field providing some geological history
- maxima in the observed terrain providing better views of the area
- areas initially marked on the global map as interesting

1.4 Mission Scenario

The following is a projected mission scenario for the earth mission. While not mentioned in this scenario, the potential for deploying a scout vehicle to gather extra information is also available to Daedalus. The information gathered by the scout could provide Daedalus with route information, feature value, or other useful information.

- Pack Daedalus for transport as described above
- Transport Daedalus to selected site
- Testing of systems to ensure proper operation of all modules
- Deploy Daedalus at random location within the site
- Initial Estimation of Position by Daedalus confirm with GPS or Human Intervention
- Begin Routine Operations:
 - Global Route Generation
 - Traverse/Observe Terrain
 - Maintain Feature List adjust global route
 - Locate Feature
 - Track Feature
 - Perform science
 - Proceed to rescue location
 - If at evening then hibernate for night
 - Have systems checked and batteries replaced
 - Wake and perform diagnostic tests
- Repeat Routine Operations until end of mission time.

- **Pick up Daedalus from final location**
- **Pack Daedalus for transport as described above**
- **Return Daedalus to CMU**

2 Configuration

One of the key elements to the success of a robotic mission is the configuration of the rover¹. To date, no systematic means for selecting an appropriate rover configuration exists. Currently, this decision is based on the experience and prejudices of the rover designer. Furthermore, the rationale used for selecting one configuration in preference to another is typically not clear. In this report, the intent is to clearly show the criteria considered and how they were evaluated.

Discussions of robot configuration frequently involve tables showing relative strengths and weaknesses of the different concepts being considered. This approach will not be used because it tends to be subjective and to create more problems than it solves. Instead, this document will focus on a particular configuration, one that was developed to meet the specific needs of the mission. Its strengths and weaknesses are highlighted and it is compared to other configurations that were considered for this mission.

Typical matrix evaluations of rover configurations consider a multitude of criteria, typically including such items as payload volume, stability, complexity, etc. However, these criteria do not really get at the heart of the issue, the ability of the rover to carry out some specified mission. The evaluation presented here will attempt to directly address this issue by examining the proposed configuration in three broad categories: its ability to be delivered to the target area, its long term reliability and its capability for carrying out a mission. Sections 2.1 through 2.3 discuss the choice of the lander/rover concept, a legged configuration and the specific walking beam configuration in light of these three criteria. Section 2.4 is an evaluation of the capabilities of the Daedalus configuration.

As stated in Chapter 1, the design of the Daedalus robot is heavily influenced by criteria for the lunar robot. Although the following discussion is a justification for the Daedalus configuration, justifications that apply to the lunar rover are used also.

2.1 Lander/Rover

The traditional means for delivering a rover to a planetary body is to deliver the rover to the planetary surface as the payload of a dedicated landing stage. Examples of this include the Soviet Lunakhod, the Lunar Excursion Module of the Apollo program and the proposed Artemis common lunar lander. The biggest problem with these systems is the inefficient use of mass delivered to the planetary surface. For example, if we consider the mass of the entire rover to be the delivered payload, of the total mass delivered to the lunar surface by the Artemis, only 15 percent is payload. (NASA hopes to increase this to 45 percent for later missions.) The proposed lander/rover concept, however, is capable of delivering a payload in excess of 90 percent of the delivered mass. In today's economic/political climate this means that the same payload can be delivered with smaller rockets, a driving factor in the cost of extraterrestrial, planetary missions.

From the above discussion, it is clear that the greatest advantages of the lander/rover concept are accrued in the area of system delivery. The lander/rover concept also possesses certain advantages in terms of reliability, but other than reducing mission costs or allowing greater capability for the same cost, the lander/rover does not present any clear advantages in terms of the ability to carry out a mission.

1. Within the context of this report, a rover configuration is taken to mean the number and the relative position and orientation of the locomotion devices with respect to the body of the rover. For example, a solid rectangular body with four parallel wheels, two on each side, separated by a fixed distance, would describe the configuration of an automobile. It is important to note that the concept of a rover configuration is independent of the size of the rover.

Delivery

- The lander/rover concept allows larger payloads to be delivered with a fixed launch vehicle since a separate lander is not required.
- The lander/rover is simpler to integrate into a launch vehicle.

Reliability

- Shared components, such as computing, power, telemetry and structure eliminates unnecessary duplication of mission critical hardware. By reducing the component count, system reliability is typically increased.
- The combined lander/rover eliminates the need to deploy the rover from the landing stage. Although the deployment maneuver does not represent a high risk, it is a single point of failure. With the combined lander/rover, failure to jettison the fuel tanks does not kill the mission.

2.2 Legged rover

The selection of the locomotor for a planetary exploration robot has the greatest impact on the configuration of that robot. To make a proper selection, not only are the characteristics of the locomotion elements compared, but their amenability to the lander/rover concept must also be considered. The three means of locomotion considered in this report are legged, wheeled and tracked. Since tracks can be considered to be large wheels, in the discussion that follows, the statements that apply to wheels also apply to tracked vehicles.

So as to make valid comparisons, the same launch vehicle is assumed for all configurations. To minimize the number of assumptions required, launch vehicle payload mass and volume consumed by the landing stage, which may be required for many of the conceivable wheeled configurations, will be neglected.

Delivery

- Legs are better suited to the lander/rover concept than wheels. The key concept behind the lander rover is the sharing of components. Although it is possible to envision landing on wheels that are suitable for planetary exploration, this would be a challenging problem, both in terms of materials and controls. A more likely solution would be to have a separate lander if a wheeled vehicle were used. It is also difficult to imagine how the descent stage could be directly appended to a wheeled vehicle because of placement constraints and ground clearance requirements.
- Legs can be used to actively decelerate the robot as it touches down on a planetary surface. Although a challenging control problem, this deceleration by the legs simplifies the requirements for the descent rocket and descent control system; a problem that may prove to be more challenging than the leg control problem. Previous planetary landers used heavy legs whose sole purpose was to absorb the landing shock.
- Legs typically provide greater ground clearance than wheeled robots. This clearance is needed for rocket exhaust and a deceleration zone to prevent the body from hitting the terrain upon landing, since the body is not designed to withstand high-g loading.
- The (typically) wider spacing of robot legs than robot wheels provides a more stable base for landing. The wider spacing is important to minimize tip-over on landing for the lander/rover concept.

Reliability

- Walkers suspend themselves over the terrain on discrete contact points, thus allowing them to avoid marginal footholds, to minimize dynamic shocks to the body, to maintain a stable pose independent of the underlying terrain and to simplify the perception and planing prob-

lems. Wheeled robots maintain continuous contact with the ground and must traverse all points along a line of travel, thus none of the listed benefits are achievable. (These arguments assume a vehicle without an active suspension, a mechanism that would greatly increase the complexity and mass of a wheeled rover, but would allow it to realize some of these benefits.) [Bares 91]

- Legs are more efficient in soft materials than wheels, due to the low shearing strength of the materials. Should soft materials be encountered, it is more probable that a legged rover could extricate itself than a wheeled machine. Consider the case of a wheeled vehicle getting stuck in sand that a person can easily traverse.
- Legs can be used to probe potential ground contact point before committing to them. By testing a footfall before using it, the probability of tip-over through ground failure is reduced.

Capability

- The ability of a legged vehicle to traverse rugged terrain is a function primarily of the rover's geometry. The ability of a wheeled vehicle to negotiate rugged terrain, however, is not only a function of its geometry, but also the underlying terrain. Thus, given a walking and rolling vehicle that can (kinematically) negotiate the same terrain, the walking robot will have greater terrain capabilities than the wheeled machine.
- Since the body of a walking machine is isolated from the underlying terrain, it can be precisely positioned. This is useful for gathering scientific data or pointing the antenna while moving.

2.3 Daedalus configuration

Based on the discussions above, the Daedalus is a hexapod walking beam robot, the simplest and most robust type of walking machine. This information is not sufficient to configure the robot. The following discussions show how this concept, a hexapod walking beam, was transformed into the configuration shown in Figure 2.3-1.

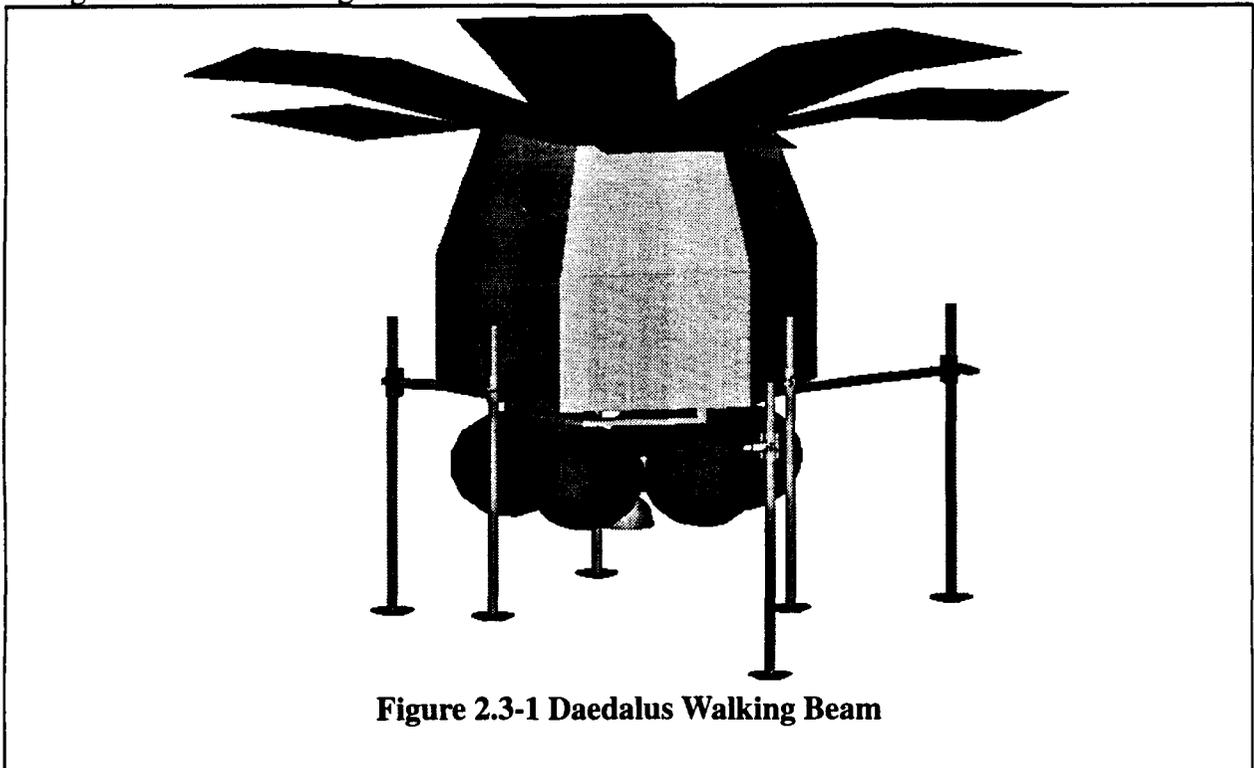


Figure 2.3-1 Daedalus Walking Beam

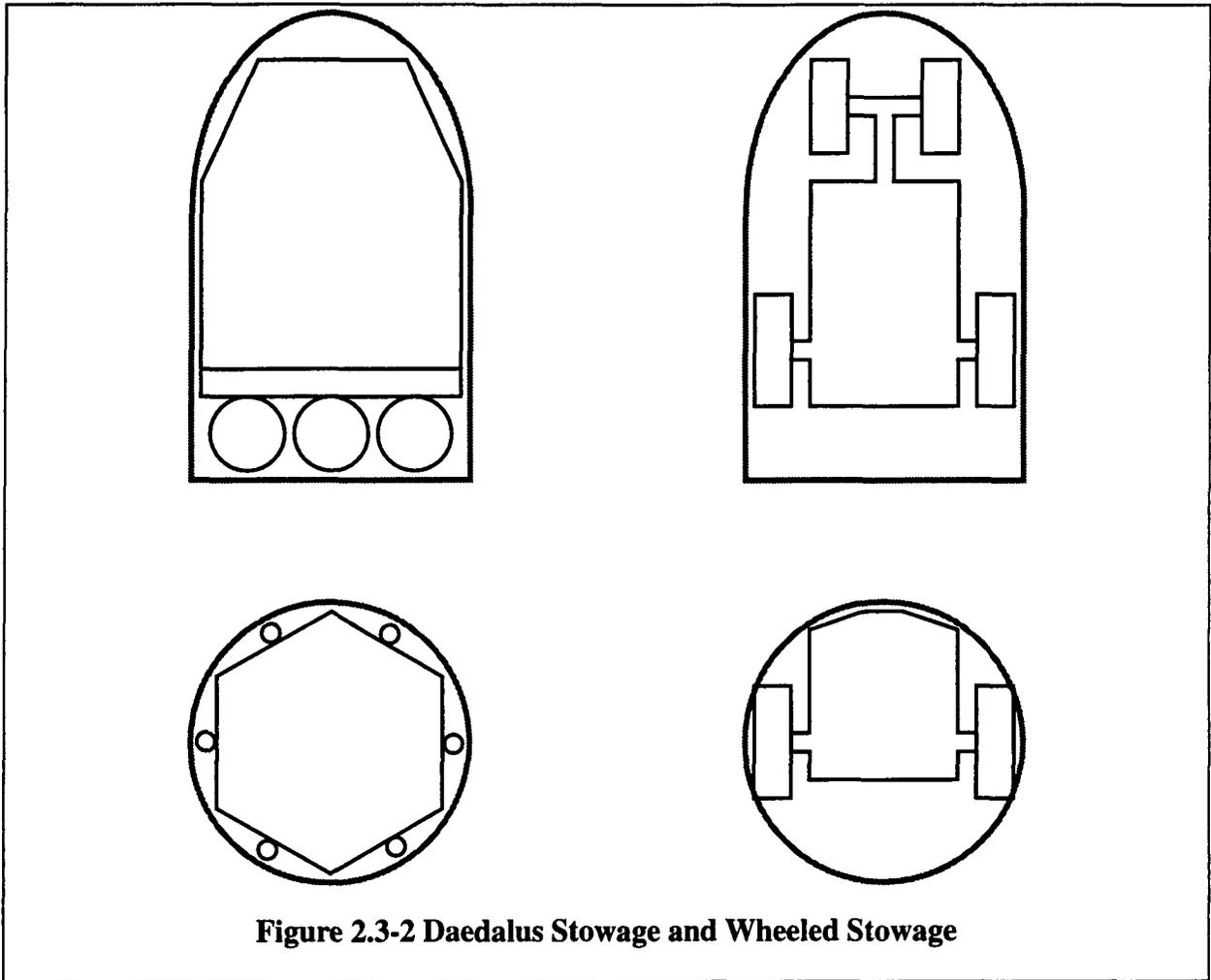


Figure 2.3-2 Daedalus Stowage and Wheeled Stowage

Delivery:

- The shape of the Daedalus was chosen specifically to maximize utilization of the payload volume. To achieve compact stowage, the Daedalus deploys a single leg after landing. Because of the Daedalus's configuration, this deployment uses the horizontal drive motor in its typical operating mode, thus no additional mechanisms or risk is entailed with this maneuver. Another advantage of the Daedalus configuration is that it scales well over a wide range of launch vehicles. Interestingly, as an Daedalus-like rover gets larger, the mass fraction consumed by the locomotion element decreases. From the perspective of the Daedalus project, this compact stowage means that the Daedalus will be easily transportable to test sites.
- Space vehicles are basically axisymmetric to optimize utilization of the payload fairing and to simple spacecraft integration. The Daedalus design is axisymmetric, but wheeled configurations are typically not.
- The locomotion elements of the Daedalus concept consume less than 25% of the mass delivered by the launch vehicle and only a very small fraction of the payload volume. A wheeled configuration with rigid wheels will necessarily consume more payload volume see Figure 2.3-2. The wheeled configuration shown is one that maximizes usable payload volume because of the front bogey wheels. Other configurations, such as a double rocker bogey with side-mounted rocker arms, will stow less efficiently. Wheeled robot stowage efficiency can be improved by using a variety of deployment devices or inflatable wheels.

However, a wheel deployment device adds complexity and weight to the design. The terrainability of large, inflatable wheels is uncertain, due to the limited choice of usable materials for this application. It should be further noted that Figure 2.3-2 is very favorable for the wheeled configuration for two reasons:

- The volume consumed by the lander and the deployment ramp are not taken into account
- The width of the wheels is probably too small to provide proper traction on soft surfaces. Note that increasing wheel width marginally has a great impact on rover volume.

Reliability:

- For a mechanical system, it is generally true that the simpler the mechanism, the more reliable it will be. As Bares states [Bares 91], “The simplest walkers that can traverse rough terrain in a statically stable manner are frame walkers”. For a planetary mission, simplicity is an absolute requirement since service calls are not possible. It has been suggested that the number of actuators can be used as a metric for complexity, however, this is not generally accepted, although others suggest that complexity is a function of the number of DOF’s [Bares 91]. For example, a single actuator could be coupled to multiple gear boxes, which in turn drive other gear boxes through a series of chains and pulleys. A system such as this is far more complex than the simple, independent, linear actuators used on the Daedalus and also introduces multiple, single-point failures.
- The Daedalus configuration is inherently stable, and the center of gravity remains within the polygon formed by the feet in contact with the ground at all times. This is not necessarily true of other walkers, see Section 2.3.1 below.
- The mechanical components used for the Daedalus are those that can inherently withstand the harsh, lunar environment without modification. The solid bearings used can withstand the temperature cycling and the vacuum of the lunar environment. Furthermore, they are specifically designed to work in dirty locations without the need for maintenance. While the bearings on the Earth-based engineering model may need to be replaced after extended operation, the duration of the lunar mission will be short enough such that bearing wear will not be an issue. The rack and pinion drive, based on our Ambler experience and from other sources, is very robust in dirty environments. The motors used for the lunar mission will be fully space qualified and are capable of withstanding the environmental conditions.
- Redundant components reduce the likelihood of non-recoverable, system failures. The robot should be designed to minimize the number of possible single point failures. This implies that using a single motor with gear trains may not be a good design practice. The Daedalus makes use of dual-redundant motors on all axes. This allows the Daedalus to keep operating, and with only slightly diminished capabilities, in the face of a motor failure. [The wheeled concept mentioned above would be crippled if *any* component in the drive train fails.] For the earth based engineering model, the redundancy takes the form of two motors attached to a cycloidal gearbox. For the lunar mission, this redundancy would be in the form a motor with dual windings.
- The Daedalus never has more than three legs supporting the body at any given time. This minimizes the potential for actuator conflict, which can be harmful to the system and consumes excess power.

Capability:

- Bares states [Bares 91], “Frame walkers are most suitable when the legs are long enough to support the frames above the surface roughness.” The Daedalus’s legs are long enough to for terrain roughness of up to one meter. Furthermore, the Daedalus design is based on

an orthogonal leg, which has many advantages for autonomous robots when compared to other types of leg.

- For the Daedalus to avoid an obstacle, a change in the nominal path of travel is required, and this deviation will typically not be in the direction of the current goal. However, small deviations from a nominal path for an extended mission will have little impact on the overall success of the mission.
- To get the most data from an exploration robot, it should move continuously and “quickly”. Although walking rovers are “slower” than wheeled robots on hard, flat surfaces, the speed advantage disappears in rugged and soft terrains. Furthermore, the limiting factor in a walking machine’s ability to traverse rugged terrain is the computing required to determine foot placement, since many of the surface irregularities do not alter the robot’s motion. For a certain wheeled machine, this is not true, because even in negotiating surmountable surface irregularities, the body is subjected to additional dynamic forces, forces that eventually limit the robot’s ability to move faster.
- The only significant drawback to the Daedalus design is that the feet can not be independently placed. However, Price and Chun [Price], who have extensive experience with this type of machine, claim that this is not as great a limitation as might be expected. In the course of their testing (albeit with human teleoperation), they have yet to be in a situation where the fixed foot pattern has limited the rover’s ability to continue making forward progress. They also noted that the walking beam was capable of traversing almost any vector through their simulated Martian terrain.

2.3.1 Comparison

The previous discussion focused solely on the Daedalus configuration. To more clearly demonstrate its strengths and weaknesses, the Daedalus is compared to three other configurations and the discussion will focus on those areas where there are differences. The three other configurations considered are a follow-the-leader walker (FTL) (a radially-symmetric FTL walker Bares 91, page 51)), the Russian Marsokhod with a descent stage and rocker-bogie wheeled rovers with a descent stage. For all configurations, the same stowage volume is assumed.

2.3.1.1 Daedalus and the FTL

The FTL allows selection of individual footfalls, can negotiate terrain that is somewhat more rugged than the Daedalus and has additional possible gait patterns, such as crabbing sideways. These advantages come with the following cost: a significantly more complex mechanism, a higher locomotion mass fraction, decreased stability and increased difficulty for planning and control. In addition, the FTL will be slower than the Daedalus.

The FTL’s advantages arise from its ability to individually place its feet. Because of this ability, the FTL can alter its gait and stance, thus allowing it to negotiate very rugged terrain at reduced speed and with diminished stability margin. However, this does allow it to explore terrain that might be impassable for the Daedalus. Its design also allows additional movements, such as crabbing sideways, which are impossible for the Daedalus. However, making use of these additional is difficult, and although the Ambler also possessed these capabilities, they have never been used by the autonomous system.

The FTL’s most serious drawback is its tremendous complexity. The FTL requires 18 actuated DOF’s and six passive DOF’s. This is three times as many motions as the Daedalus design. These additional DOF’s require additional hardware, thus increasing the mass of the robot and decreasing the useful payload. These additional DOF’s also increase the difficulty of the planning and control problems, specifically, more legs need to be planned for and coordinated motions are required.

Although the Ambler concept was pursued in favor of the FTL concept, due to the perceived advantage of the circulating gait, Bares showed that the stability of the FTL is greater than that of the Ambler. Although the stability of all walking machines depends on the software's functioning properly, the Ambler and the FTL have a software-related stability failure mode not present in the Daedalus configuration.

Finally, the FTL will be slower than the Daedalus. For the Daedalus to move one body length forward, a set of legs is picked up, the body moves and the legs are lowered. For the FTL to move one body length forward, each of the legs must be moved forward (this is most quickly done using a tripod-tripod gait), then the body is moved forward in a coordinated move. Depending on the relative dimensions of the body and legs, this cycle might have to be repeated more than once. The FTL can certainly move as quickly as the Daedalus, however, to do so would require significantly more power than the Daedalus consumes for a similar move.

2.3.1.2 Daedalus and the Marsokhod

The Marsokhod exists, thus enabling real performance evaluation. Claimed advantages, such as terrainability, ease of control, stability and simplicity are open to discussion. Disadvantages of the Marsokhod include: lack of amenability to the combined lander/rover concept, minimal payload volume which is divided among three chassis, a significantly more complex perception problem and minimal ground clearance.

The author's observations of the Marsokhod during the testing in Death Valley have raised some serious concerns about its capabilities. First, it is claimed that the machine is capable of negotiating rugged terrain. During the testing, it climbed slopes of not more than 15 degrees and climb boulders less than 0.25 meters tall while on the slope. While performing these maneuvers, progress was quite inefficient and the Marsokhod was under direct human control. The ability of a wheeled vehicle to climb a slope is a function of the shearing strength of the materials, thus, on soft materials, a wheeled vehicle will be severely limited in its ability to handle steep terrain. Furthermore, after a run of not more than 150 meters, technicians straightened the wheel flanges and generally tinkered with the machine. Second, although controlling a wheeled vehicle is simple, control of the Marsokhod peristaltic motion certainly is not. Not only does using this capability require two independent control modes for the rover, the planning software must also be able to determine when to use this mode and when to use the default mode. Third, it is claimed that the Marsokhod is a very stable platform. While this is certainly true in the direction of motion, the dimension of the Marsokhod perpendicular to its direction of motion is quite small, thus reducing stability. This would be a major cause of concern during a transverse slope traversal, since sideways slipping of the robot will decrease the energy stability margin. Finally, it is claimed that the Marsokhod is a simple device. However, the Marsokhod has eight actuated DOF (the same number as the Daedalus) and two passive DOF. However, wheel drive mechanisms are typically simpler than the Daedalus's linear drive systems.

It is difficult to conceive of a simple way to make the Marsokhod amenable to the lander rover concept. The Marsokhod will require a separate landing stage from which to deploy. Giving this concept the benefit of the doubt, a landing stage will consume at least 20% of the available launch vehicle mass and at least 20% of the launch vehicle payload volume. This, coupled with the large width of the Marsokhod wheels, will result in an unusably small volume for computing and scientific payload if a small launch vehicle is used. Although this concept might work for a Delta class launch vehicle, or larger, the mission costs would be prohibitive. Another difficulty is that the Marsokhod payload volume is split among three separate chassis. Thus greatly complicates the environmental conditioning problem and will require extra mass.

Another difficulty with the Marsokhod is that, like all wheeled vehicles, the body follows the underlying terrain. This greatly complicates the perception problems because it becomes hard to

predict the subsequent image. Also, should map merging be required, this becomes very difficult. Another difficulty with this design is the difficulty in pointing an antenna while moving. To do so would require a significantly larger antenna pointing system, more power and a fairly complex control algorithm. In all likelihood, the Marsokhod would have to stop to transmit data back to the Earth. Considering data bandwidths and the amount of information to be gathered, this might necessitate being stationary for 50% of the time on the lunar surface.

Finally, the Marsokhod has little ground clearance. This cause two difficulties. The first is that the perception system must be able to detect small object and planning algorithm system will be constantly generating routes to avoid objects that are insignificant to the Daedalus. The second is that the Marsokhod will be required to frequently traverse obstacles, a maneuver that is risky and consumes excess power.

2.3.1.3 Daedalus and rocker bogey configurations

It is difficult to make many incisive comments regarding the rocker bogey concepts because little has been published about existing rocker bogey rovers. Furthermore, those that do exist are much smaller than the size of the Daedalus and their ability to scale to larger sizes is unknown. In addition, these concepts appear to have several difficulties, including lack of amenability to the lander/rover concept, poor stowage in a launch vehicle and complex drive systems that require multiple gear boxes, clutches, etc. However, these concepts provide some of the benefits of legs, such as terrain isolation, with some of the advantages of wheels, such as speed.

2.3.1.4 Summary comments

It is said that wheeled vehicle are faster than legged ones, for given power, and for hard flat surfaces this is certainly true. One soft surfaces and in rugged terrain, this speed advantage is diminished. For the missions under consideration, the speed difference on benign terrain is not so clear. For a given launch vehicle, the size of the wheeled vehicle delivered will be significantly less than the size of delivered Daedalus. Since the vehicle are quite different in size, the larger Daedalus might be capable of the similar speeds as the smaller, rolling vehicle.

In summary, although the FTL configuration offers some benefit in terms of terrainability, these advantages are not sufficient to overcome the added complexity of this system. The Marsokhod seems to offer no technical advantages that make it a viable alternative. The rocker bogey configurations also appear to offer no technical advantages although they need to be studied in greater detail.

2.3.2 Design considerations

In this section, several issues that have been raised as concerns will be addressed:

- Exposed drive mechanisms - The Daedalus drive mechanism is a motor with a pinion driving a rack. The comments that follow pertain to the flight article, although the argument applies to the engineering model. The rack is mounted to a tube that serves both as the load bearing member and the bearing shaft, see Section 3.1.2. The bearing material is Frelon, a Teflon compound that is bonded to the aluminum structure. This material has an operating temperature range of -400° to 500° F, does not outgas in a vacuum and is designed to work in harsh environments. For example, these bearing are used in factories, where they are subjected to very small particles of harsh dust. A part of the lunar program will be to subject the actuator system to simulated lunar dust in a vacuum chamber and to quantify the affects on the system. It is also known from experience that a rack and pinion drive is quite rugged in the face of dust. Thus, few problems are expected from the exposed drive mechanisms.
- Number of legs on the inner chassis - The Martin Marietta frame walker is configured quite differently from the Daedalus. In their configuration, the size of the inner body limits the stroke of the translational actuator. To maximize this stroke, the body size was made small,

and a fourth leg was added for additional stability. The Daedalus configuration does not require a fourth leg on the inner frame because the spacing is the same as for the outer frame.

- Stiffness and backlash - The Daedalus is being designed to minimize these problems by accounting for them *a priori*. By using conservative estimates of the robot mass, the various members are being sized to deflect only a certain amount when under load. The technique being used to estimate these calculations is also quite conservative, and actual deflection will be significantly less. Backlash in the mechanisms is being handled in a similar manner.
- Placement of scientific instrumentation - We can not know, without selecting a suite of instruments, how to place them in a robot. However, the Daedalus provides the largest possible payload volume for scientific instrumentation of any of the candidate configurations, thus minimizing this problem.

2.3.3 Power, stability and ground-clearance

An open area of research is the trade-off between power expended for locomotion, stability and ground clearance. For maximum stability, the body of the robot should be kept as close to the ground as possible, however, this is power inefficient as frequent body lifts will be necessary for obstacle avoidance. For minimum power consumption, the body of the robot should be kept as high as possible, thereby guaranteeing that a body lift maneuver need never be done, however, this minimizes the stability of the rover. The optimal body position probably lies between the two extremes, and is probably a function of the distribution of the obstacles to be encountered and the general roughness of the terrain.

The energy stability margin (ESM) is defined as the least amount of energy required to raise the center of mass of the robot over one of the edges of the support polygon. For the Daedalus, this condition is shown in Figure 2.3-3:

As might be expected, the ESM depends on the height of the center of mass in the body. Thus, by placing the center of mass as low as possible in the body, the ESM is improved. To calculate the ESM of the Daedalus, several assumptions are required. These are:

- The mass of the y-frame assembly is $0.12M$, where M is the total mass of the Daedalus.
- The center of mass of the y-frame is located $0.66L$ from the right side and $0.3L$ above the bottom of the y-frame
- The center of mass of the body is located $0.2L$ above the bottom of the body.

The combined center of mass of the Daedalus can be calculated to be $0.21L$ above the bottom of the body and $0.08L$ to the left of the body center line. The bottom of the body is $0.92L$ above the ground and the tip-over feet are $0.25L$ to the left of the body center line. Thus, the ESM is calculated to be $0.01MgL$. This calculation is the worst case ESM calculation, all real poses will possess greater stability. Thus, 26J are required to tip the Daedalus over.

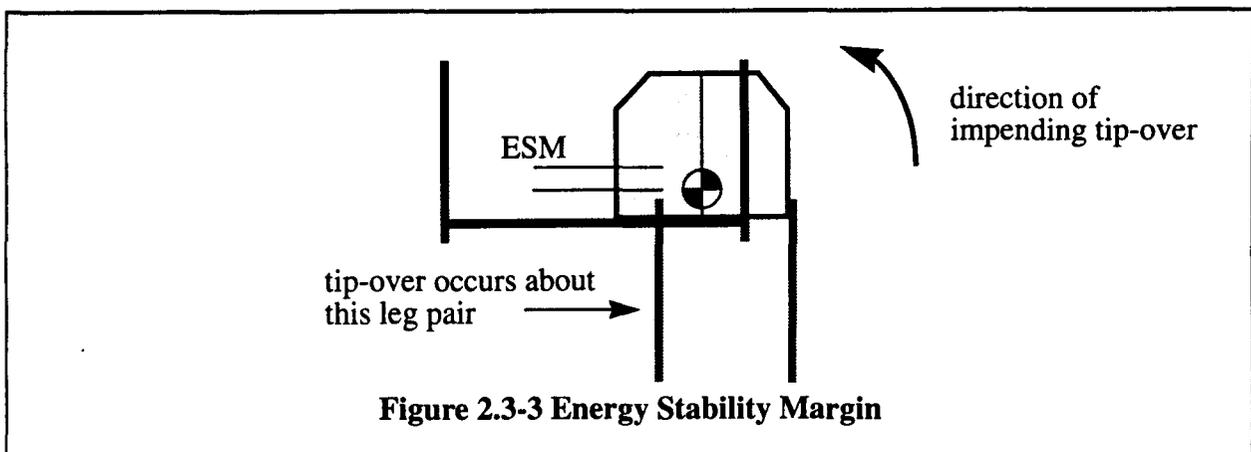
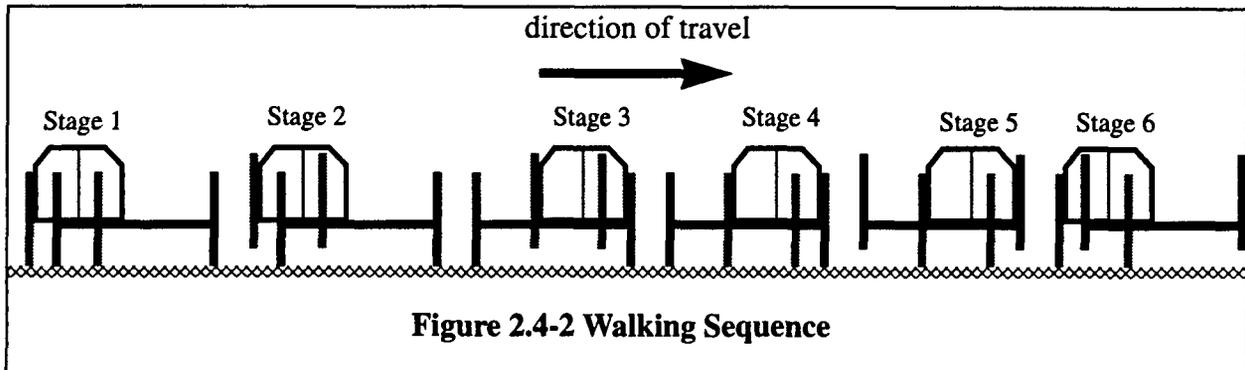


Figure 2.3-3 Energy Stability Margin



2.4 Daedalus capabilities

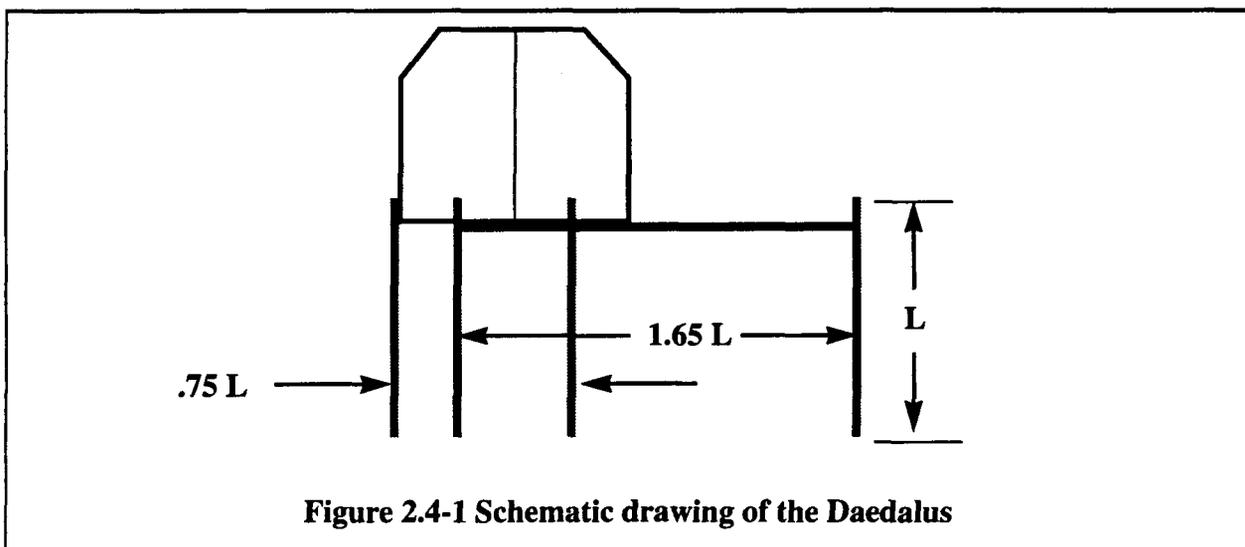
This section describes the Daedalus walking cycle and the Daedalus's kinematic capabilities. Figure 2.4-1 below is a schematic representation of the Daedalus. The dark legs are those attached to the y-frame and the lighter legs are those attached to the body. The horizontal actuator has a stroke of approximately $0.9L$ (and not L , due to the volume consumed by the mechanism). The vertical actuator has a stroke of approximately $0.93L$ (and not L , again due to the volume consumed by the mechanism).

2.4.1 Walking cycle

A complete cycle of motion for the walking beam is comprised of six distinct stages, where the transitions between the stages are called phases, with phase 1 being the transition between stage 1 and stage 2, see Figure 2.4-2.

2.4.2 Slope traversal

One of the design trade-offs for a walking beam is the length of the y-frame. For a fixed leg length, a longer frame enables longer strokes, but also limits slope traversal. For the Daedalus, the length of the y-frame was dictated primarily by stowage considerations, so the slope traversal ability must be evaluated based on this given dimension. For the Daedalus, slope traversal is defined to be the steepest slope the rover can traverse while maintaining full stride. The Daedalus's motion is restricted in that the body must remain level at all times. Figure 2.4-3 shows the Daedalus traversing the steepest slope possible.



Two interesting facts come to light in Figure 2.4-3. First, the magnitude of the maximum slope traversal is determined from the body fixed legs, not the y-frame fixed legs. The reason is that the vertical stroke of the forward (in the direction of motion) leg is not fully utilized. Second, either set of legs can be used to provide the vertical motion, assuming that the horizontal impact of the feet with the terrain can provide a stable foothold. This implies that a crippled leg may not greatly impact slope traversal capabilities.

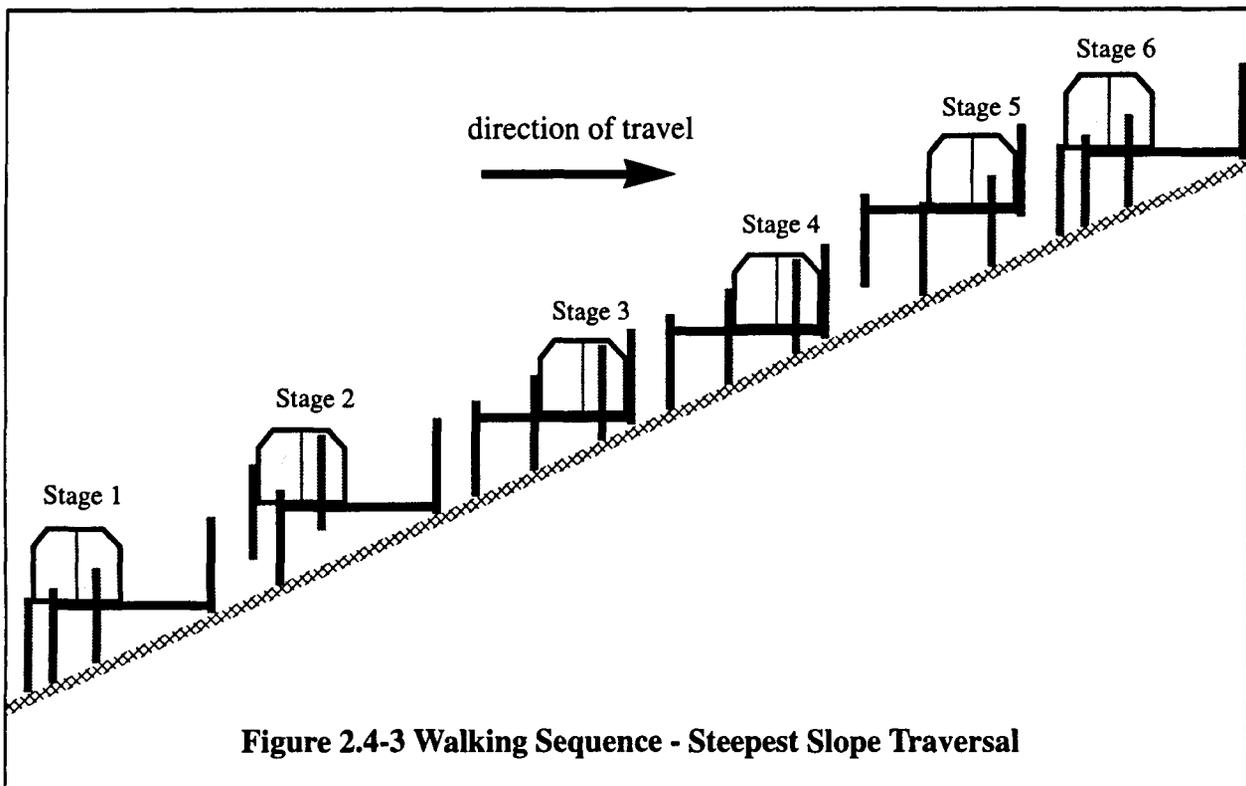
Using the dimensions from Figure 2.4-1, the greatest longitudinal slope the Daedalus can traverse is approximately 25 degrees. (Note, the Earth-based Daedalus is designed with slightly longer legs than those shown in Figure 2.4-1. It will be capable of traversing slopes of approximately 30 degrees.)

The greatest transverse slope traversable is simply a function of the leg length and the body width. For the Daedalus, the greatest transverse slope traversable is in excess of 40 degrees. [Note, the actual Daedalus is designed with slightly longer legs than those shown in Figure 2.4-1 and transverse slope traversal will be in excess of 45 degrees.] The Daedalus is capable of traversing its maximum longitudinal and transverse slopes simultaneously.

2.4.3 Step and ditch crossing

The Daedalus can negotiate steps of $0.93L$. Figure 2.4-3 shows the Daedalus negotiating the tallest possible step. This sequence assumes that the material at the edge of the step is solid and able to support the loads placed upon it. In addition, it must be noted that the Daedalus requires a ledge behind the step that is approximately $1.75L$ deep.

The greatest drawback to the Daedalus design is its limited ability to negotiate deep ditches, where deep means that the Daedalus's legs can not reach the bottom of the ditch. Figure 2.4-3 below shows the sequence for ditch crossing. The widest ditch than can be crossed is $0.5 L$ wide.



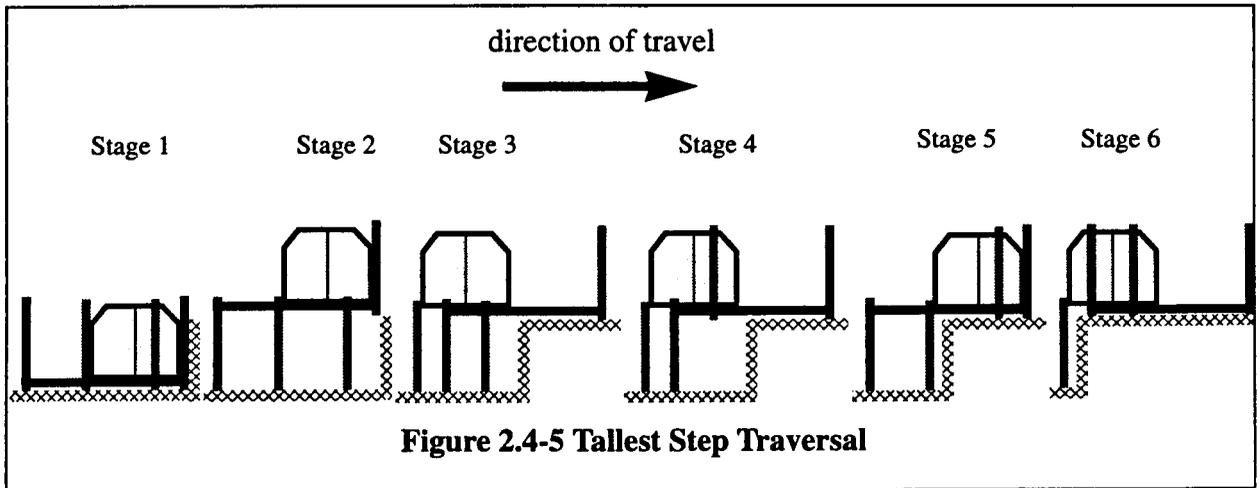


Figure 2.4-5 Tallest Step Traversal

To perform this maneuver, several shortened steps are required. Like step climbing, ditch crossing also requires that the material along the edge of the ditch can support the applied loads

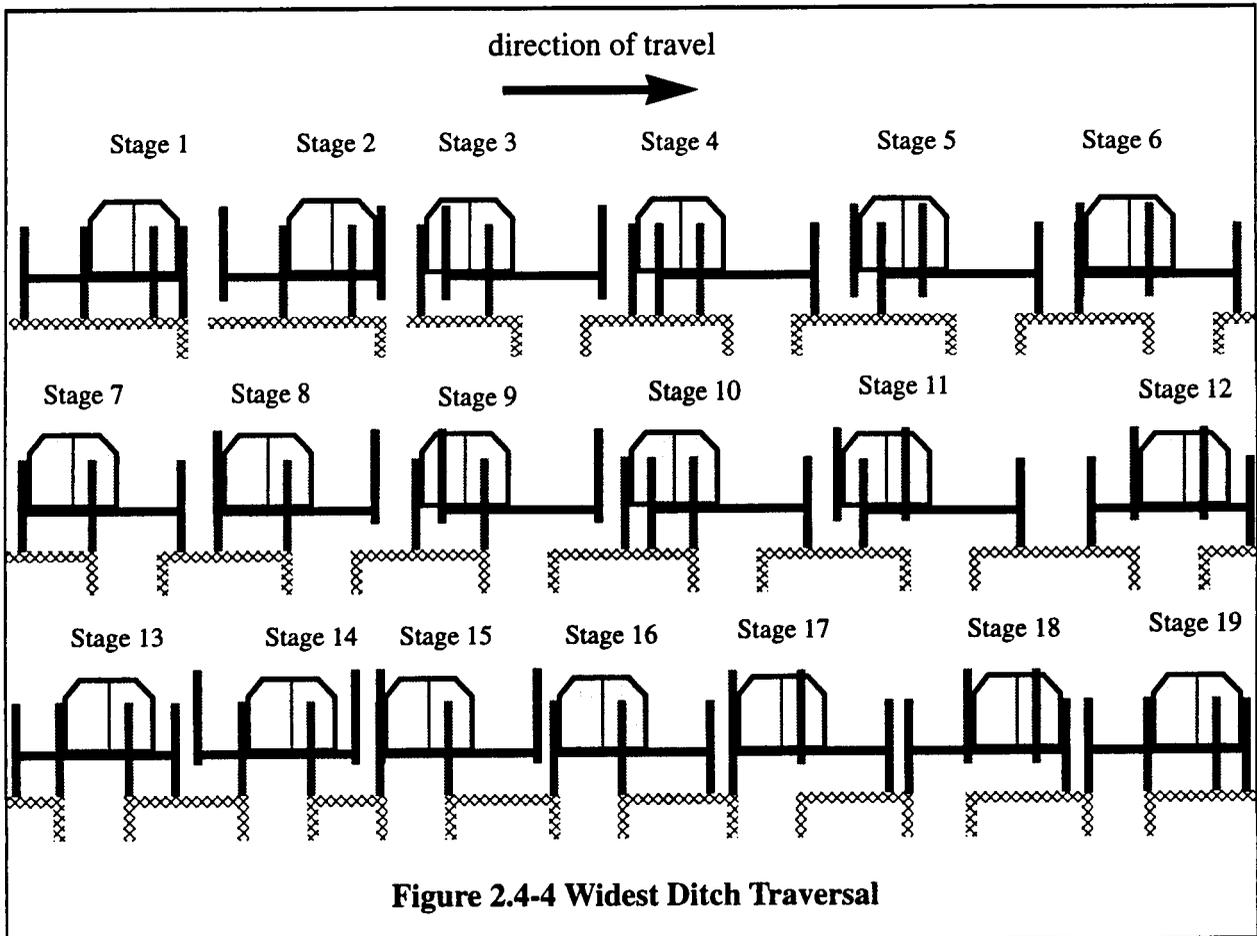


Figure 2.4-4 Widest Ditch Traversal

2.5 Scout

2.5.1 Motivation

In the APEX mission, we originally envisioned a robotic inspection of the lunar or earth surface for lava tube openings. Now, we have revised the mission to be a more general lunar exploration. The Daedalus robot is a six legged frame walker capable of traversing significantly varied lunar or terrain desert terrain. However, for the original mission, a secondary goal was exploration of the *interior* of lava tubes once they are found, and serious doubts were been raised about the Daedalus design's capabilities for safe and efficient descent into the lava tubes, exploration of the interior, and ascent back to the exterior surface. In the course of examining this problem early in the APEX project design, we came up with the idea of a scout robot, and now we find that this idea is appropriate for the more general mission, too.

2.5.1.1 Project

Here, we have studied the feasibility of a secondary body for the robotic system. This secondary body would be capable of descent into the terrain unfit for the Daedalus framewalker and sensing and exploration consistent with the demands of these hostile environments. The scout body would be a light and simple design streamlined to the task of quick extremely rough terrain exploration.

2.5.1.2 Shortcomings of the Daedalus design

We have tried to establish the bounds of what navigation we can and will want to attempt with the Daedalus body in order to define the direction of the scout project. This has involved examining geological papers, consulting geologists and space scientists, and examining earthbound lava tube sites.

One of the factors leading us to consider a scout robot is that the slopes of the edge of the rift caused by the lava tube might be very steep near the mouth of the intact tube. On the earth, the slope can be more than 45 degrees while on the moon a standard slope might be as much as 60 degrees which is clearly outside the operating capabilities of Daedalus. Although it is often possible to find a gentler slope away from mouth of the intact tube, we cannot guarantee this and even if we knew a gentler slope would exist, it could be a considerable distance from the mouth (worse so on the moon than on the earth even since all geological formations are an order of magnitude larger on the moon). Furthermore, the moon's surface is covered by up to 10 cm of dust and or *small shardlike rocks*, and the large Daedalus framewalker might not be able to establish firm footholds on the slopes leading down to the mouth on such soil. Sliding is "very likely" (according to Dr. Cassandra Coombs).

Additional obstacles near the mouth of the intact lava tube include dense vegetation on the earth and significant rubble on the moon which could make it impossible for Daedalus to enter into or even acquire any sensor data from the interior of the lava tubes. All of these concerns lead us to the point of examining different scout configurations to see if the additional cost of the scout is worth the scientific gain or even if we can negate the cost in savings on power and weight on the framewalker. These concerns carry over directly to any interesting lunar feature which we would want to explore, especially to craters.

2.5.1.3 Goals of scientific study

We need to firmly establish what information we seek to acquire in the lava tube exploration or other lunar mission. This information constrains both the sensors which we wish to put onto the scout and the mode of locomotion. Some reasons we might want to explore the interior of the lava tubes (as per the original mission) follow. this is to be used as an example of what might be needed in any lunar exploration mission.

- **Pristine Environment For Samples for Chemical Composition & Aging**

The interior of an intact lava tube is uncontaminated by meteoric materials. From a geological standpoint, this means that if we wish to study the composition of lunar materials, the interior regions of lava tubes may be the most accessible location to avoid recent meteor fall-out. Clearly, due to the nature of their formation, lava tubes present us with a certain type of lunar material (volcanic in origin), but even getting samples of this type of material will yield useful information about the origin of the moon and what we might be able to eventually mine from the lunar grounds

- **Geological Layering**

Inside the lava tubes there should be geological layering visible in lava tube walls. This layering would provide scientifically valuable information about different stages of the moon's geological development and different stages of vulcanism.

- **Vesicles**

Vesicles, or bubbles in the rock of the lava tube walls, provide a history of volatiles. These should be able to be seen visually, and their changing size and orientations (since they would typically be ellipsoid) direct us towards the source of vulcanism. The source of the lava flow is not always readily apparent on the moon where many of the volcanos have long since been flattened by meteorites and subsurface lava sources are not immediately obvious. Locating the source of lava would, clearly, direct use towards other potential lava tubes

- **Crystals**

Crystals in the rock which might be visible also tell us about the direction of the source of the lava flow. One earth the larger crystals would be found nearer the source of the lava flow.

- **Thermal Stability**

Scientists suspect a remarkable thermal stability in the lava tubes, which is a highly desirable characteristic on the moon where there is such an enormous change in temperature from light to shadow. By gaining sensor access to the interior of the lava tubes, we might see whether the thermal stability we desire for future missions is actually present.

- **Large Natural Caverns**

Since one intent of the mission is to examine these lava tubes for suitability for human habitation or storage of large man-made objects, we would want to explore the "architecture" of the tubes. Floor maps will let us decide whether to further invest in lava tube exploration by seeing if sufficient space for mission purposes exists.

The following is a short list of sensors we might wish to use in the interior of the lava tubes or in other missions:

- **Video Camera**

A small CCD video camera could be mounted on any scout design. Cameras are commercially available which have << 1kg mass and measure several cm in length by 1 cm radially.

This sensor could be used for navigation and for scientific purposes. Low bandwidth communications with the framewalker would allow us to slowly transmit high resolution images for scientific use.

- **Infra-red Video Camera**

An infra-red video camera would allow us to scan large areas for temperature readings. Such images would give scientists thorough information about thermal stability within of lava tubes.

- **Thermometer**

Although the infra-red video camera would clearly be useful to and interesting for scientists, if we could simply measure temperature at a single spot inside the lava tubes or about any other lunar feature, we would know if there was any protection against the radically varying lunar surface temperatures. Comparing measurements at a few nearby discrete locations would provide information about internal thermal stability,

- **Mass Spectrophotometer**

An instrument such as a mass spec provides data on the chemical composition of rock samples. A mass spec such as one might find in a chemistry lab might weigh approximately 12kg and could measure 20cm x 20cm x 10cm. Special purpose devices could possibly be made lighter and smaller. The downsides are that these devices typically are very expensive (~100,000 for a non-special purpose mass spec) and can draw a fair amount of power.

- **Rangefinder**

A sensor which could be used in mapping lava tubes, navigating for the scout, and aiding in the framewalker's navigation would be a spot laser rangefinder. Rough dimensions for any enclosed area such as a cave or tight valley or crevasse floor could be measured very quickly, and reasonably precise measurements of the scout to an object (or even the framewalker to an object - see recommendation section) could be easily had. A single unit would run from \$7k to \$10k.

- **Geiger Counter**

In order to determine if a lava tube is fit for human occupancy, we would like to know if there is radioactive material inside. For this, we could use some form of geiger counter. If we had a sophisticated enough device, we could couple it with other sensors (mass spec & video) and potentially determine more about chemical composition and the age of the lava tube or any surface rocks we find by measuring radioactivity.

As we wish to de-emphasize lava tubes for the lunar mission, and look at all interesting lunar features the scout needs to be generalized. This would mean a less specialized set of sensors, so our sensors must be of use outside of the original scout mission. The supplementary missions of general lunar feature investigator and framewalker navigation aid (see below in Conclusions) must weigh strongly in our recommendations.

2.5.2 Possible Scout Designs

The sort of missions we can perform with a scout are constrained as much by the mechanical design as by the sensors. Here we shall examine and evaluate several possible mechanical designs for the scout. Constraints on the design include mission performance, cost, weight, retrievability, simplicity of mechanism and computing, and reliability in rough and unknown terrain. Several different possible locomotive options are examined in the section.

2.5.2.1 Tethered mechanical ballistic robot (TMBR)

The TMBR would be some sort of cushioned sensing unit which could be flung mechanically away from the robot body and pulled back via a light and strong cable. Communication to the Daedalus unit could either be via radio or through direct connection upon TMBR recovery through the tether. Motivation for such a locomotion and frame would be cost and mission specialization. That is, in both fiscal and design costs, this is a cheap locomotion modality. With a very specific sensing mission in mind, we could afford the narrow measurement possibilities this offers us. For instance, if we really most wanted to know if the interior of a lava tube was, in fact, cooler than surface temperature on the lunar day and warmer than surface temperature the lunar night, we might very well be content to stick a thermometer inside a cushioned container and fling it into a lava tube. Thus, in a situation where Daedalus is unable to progress by itself due to excessive slopes or collapsed entryways, we might still have enough of an opening for a TMBR scout. Sensors would be limited and would have to be impact resistant.

Propulsions would be from a spring packed on earth (for disposable sensors), or a repackable spring system or off of an arm or movable mast.

2.5.2.2 Tethered self-propelled ballistic robot (TSBR)

Similar to the TMBR, the TSBR would differ in mode of propulsion. The TSBR would have some sort of solid rocket fuel and could achieve greater accuracy and range. While we might find ourselves even more restricted as to which sensors we could put on the TSBR, we gain mission flexibility. If the robot were resting on a ledge of a collapsed lava tube, we could aim and fire more reliably into a lava tube opening. We also could use such a sensor for other lunar terrain features, such as to place sensors up on small plateaus or over the rims of craters which Daedalus could not scale.

The propulsion would be non-reusable in all probability, so we would either need “disposable” sensors or a slightly complicated reloading mechanism. With the high speed, straight-line (of a sort) path of travel of the TSBR, we might even be able to mount a disposable video camera on the end and take repeated images while traveling and later process these for stereo imaging.

Noting the de-emphasizing of the lava tube portion of the mission, we see restricted options on the tethered ballistic scout ideas, since these are fairly specifically tailored to the lava tube concept (in which we merely need to move the sensors over, above, or around obstacles and into the interior of the tube). This makes us examine the following more general options:

2.5.2.3 Wheeled insect robot (WIR)

The small (relative to Daedalus) WIR would self-propel via wheels of some design (possibly large inflatable wheels). Communication with Daedalus could be established through short range radio or even with a light tether, although past experience with cables says that that would be suboptimal. Some large degree of autonomy might be required, but computing on Daedalus could be

exploited as could the more sophisticated sensing on Daedalus. A beacon on the Scout would relay position and other sensing data back to Daedalus for higher level evaluation.

Some statistics from a straw man design for Mesur robot from JPL based on the Rocky series follows:

- **LOCOMOTION:** 8 wheels on 4 bogies, 4 motors (.5 watts each at full speed)
- **POWER:** Driving time per day/power budget: 2 hours at 2 watt average power (not including stop & think time)
- **MASS:** 7 kg max. Want 4 kg for better sand mobility
- **SIZE:** (stowed chassis: LxWxH) 60cm x 46cm x 30cm
- **SPEED:** 1.6 m/minute
- **TURN:** 24 degrees/s

The Mesur straw man is to be a self-contained planetary explorer. It is to carry sufficient batteries for seven Mars day/night cycles. A WIR such as the Mesur design offers us several useful capabilities. Primarily we gain speed; although the speed listed above is not very fast, with much of the computing on Daedalus and with a new role as a supplementary robot body (or remote sensor platform), we could likely move very quickly with a wheeled scout configuration. Other wheeled robot designs are potentially capable of even greater speeds. Most importantly, we get that speed for small power consumption (2 watts versus a minimum of 78 watts for Daedalus' motion). Thus if we were to explore a large section of a crater or other largely open lunar geological construct, we could have the Daedalus framewalker moving at its slower pace carrying the high-powered sensing and monitoring the less sophisticated WIR which could dart about and aid in rapid mapping and exploration.

The terrainability of the Mesur robot is unclear. At the least, this robot would be able to move about a dusty planetary plain and clearly it is maneuverable enough to avoid large obstacles. We have no readily available statistics on steepness of slope over which it could traverse, although we hope to get access to such information. It is not clear that such a wheeled robot would be able to traverse rubble which would stop Daedalus.

2.5.2.4 Legged insect robot (LIR)

Similar to the WIR, this robot would be legged and consist of one body segment or several flexibly connected legged body segments. The primary advantage to be gained from using a legged scout is terrainability. We can assume that a properly designed legged robot would exhibit some climbing abilities for going up and down steep slopes and conquering small obstacles like rocks and boulders. A legged robot would also be significantly more flexible in rough terrain in which careful foot placement is required. What we lose with a legged robot is speed over easier terrain and simplicity of design and control. A wheeled robot could exhibit better understood dynamic characteristics on smooth terrain than a legged robot on the easier terrain and can be mechanically simplified in some rather straightforward ways (e.g. using a drivetrain to power several wheels instead of having each motor separate). Wheeled vehicles have been used and studied for quite a long time. With a legged robot we might have to reinvent the wheel, so to speak.

In order not to have to start from scratch, let us examine an LIR which has been designed with the mission of planetary exploration in mind. Consider "Hannibal," twin to MIT's well known 6-legged Mobot (short for Mobile Robot), "Attila." Hannibal and Attila come from MIT's Mobot lab which is run under the supervision of Rodney Brooks. Although a good deal of the attention they get stems from Brooks' subsumption architecture and its application to these small robots, we are largely concerned with the mechanical specifications.

Cynthia Ferrell, a graduate student at MIT has provided the following specifications for Hannibal based upon her own observations:

- LOCOMOTION: 6 Legs, 6 motors. Electrically powered by on-board batteries.
- POWER: 2.5 watts for the electronics, 15 watts for walking
- MASS: 2.73 kg.
- SIZE: (L x W x H) 36cm x 39.4cm x 20.3cm
- SPEED: 3 m/minute *with* computing

The task of Hannibal, according to Ferrell, is “merely to wander over rough terrain.” She sums up the current performance level of Hannibal as:

...it does the basic rough terrain stuff likewalking over small obstacles, walking around large obstacles, backing away from cliffs, walking over gaps, walking up inclines, walking down declines, probes for footholds, etc. It's basic locomotion control is heavily inspired by insect locomotion research, so it displays many aspects of insect locomotion. For example, it changes speed by using a variety of gaits, adapts its gait to handle broken legs, turns with various radii, changes direction, etc. However, I feel the most significant contribution of this project is the fault tolerance capability. The robot adapts to sensor, actuator, and leg failures to minimize their effect on system performance. Obviously the goal is for the robot to effectively perform its task for long periods of time despite physical damage or failures...

This is a very good start towards the capabilities which we would like to see on the scout robot, although we would clearly want to add some sort of navigation. However, if we were to establish some sort of remote data link to the Daedalus framewalker, we could take advantage of its computing resources and leave much of the navigation off board of the scout robot. Thus, we would only need to add to the Hannibal based scout a capability for path following or, less radically, directional wandering (i.e. “head roughly in this direction”). The computing on board Daedalus could even create some sort of gradient field through which the LIR could navigate.

Ferrell further claims that Hannibal is unduly limited mechanically. Careful redesign (even starting with something as simple as lengthening the 6.25 cm stride) would greatly increase speed and terrainability. The researchers at MIT are even considering hydraulic and pneumatic controls as part of an alternative design.

2.5.2.5 Snake

The most radical design we could reasonably consider for a scout robot would be a snake. A robot which moves with snakelike motions (of which there are several different varieties) could be effective in navigating through loose rubble which is both unstable and surface-complicated (very few contiguous flat surfaces). A long flexible body made from short stiff links could potentially fulfill the mission objectives. With a camera mounted in the front we also achieve a high degree of sensor freedom. Communication could be via radio, a tether, via direct electrical connection upon reunion with the main Daedalus body.

Motion on a snake is via “directional friction.” The part of a real snake which contacts the surface over which we intend to locomote is highly frictional against the direction of motion and has very low friction in the direction of motion. This effect is achieved biologically with scales. A mechanical snake could accomplish this with scales, certain types of tape or rubberized treads, or with casters.

There are several possible motions we could implement. A wormlike motion is one in which we expand the body lengthwise and then contract. As we expand the body, the directional friction causes the front end of the snake to "grow" forward. When we contract, the rear of the snake is pulled forward also. This is a motion commonly found in worms and is called "Rectilinear Progression". Snakes exhibit three other motions commonly. "Horizontal Undulatory Progression" is what we would think of as normal snaking. We can envision this motion as sending a sine wave through the body of the snake with the x axis being the length of the snake and the y axis being its lateral displacement. A wave like this acts with the directional friction the result in a forward motion.

Another common motion in a snake is sidewinding. Sidewinding is used in low friction environments. In this motion, the snake lifts its body segments as it flexes so that only a small fraction of the body segments are touching the surface at any one time. This increases the mass above each contact point and allows the snake to move over terrain such as open areas of sand.

The final common snake motion is "Concertina Progression," which is the sort of motion a snake in a tight walled area or pipe might use. This is a successive flexing and straightening of the snake, and is slightly similar to the Horizontal Undulatory Progression, except the sides of the snake are used as contact points with the tight walls of the environment.

With these four motions, a real snake exhibits the kind of terrainability which we would desire in our scout robot. We have straightforward motion over flat terrain in the normal snaking motion. We have rough terrainability in sidewinding, and we can locomote through tight obstacles such as might be caused by collapsing lava tubes or the meteorite fallout.

In a mechanical design, snake systems can be modelled as mass-spring systems. Gavin Miller, among several others, has modelled snakes extensively in simulation with this sort of modelling and has captured snake motion quite effectively. He constructed a radio controlled model which operated on a flat planar surface such as a floor, and theorized that on uneven terrain motion could be achieved with vertical as well as horizontal waves being propagated through the robot snake. This, of course would require vertical as well as horizontal actuators, but instead of placing two such actuators at every joint, he thought the design could be simplified by alternating vertical and horizontal actuators by body segment. A similar effect could be achieved by placing horizontal actuators at the front of each segment and a vertical actuator at the rear of each segment.

Since a drivetrain causes friction at the joints when they are in any position but in a directly aligned neutral point, he used separate motors at each joint. The snake was battery powered and used fairly low power and extremely cheap parts.

Were we to invest money and research into the snake locomotion, we would gain a very terrainable scout with great mission flexibility. Sensors would be limited to long and narrow, but that describes video cameras and spot laser rangefinders just fine. Cameras could be mounted on both ends and reversible directional friction pads (or scales) could be used in order to allow backing out of tough spots (and stereo vision could be achieved that way if we desired by twisting both ends to point in the same direction!).

No hard numbers exist on power consumption or speed achievable with a mechanical design of the sort we would desire since no such mechanism exists at the present time.

2.5.3 Conclusions

Due to the revised mission which de-emphasizes the lunar lava tube search, the original need for the scout robot has diminished somewhat. However, we still see a scout robot as a useful and cost efficient tool in the Daedalus design. Therefore, we recommend the implementation of a WIR legged scout robot.

2.5.3.1 Legged Scout

The proposed scout would be based on the Hannibal mechanical design (MIT) and much of the Hannibal control scheme even with its subsumption architecture overtones. The reasons for this are that the design and control scheme is proven at least in an early prototype phase.

The mission of this scout is twofold. First, we still want the scout to explore regions which we cannot reach with Daedalus. This would include lava tubes, craters, crevasses, and any other terrain over which the heavy and not so finely positionable Daedalus could not safely travel. We would expect to be able to go up and down greater slopes with the scout and over slopes with looser dirt and rock shards underneath without causing shearing. The scout would also be able to navigate through openings in walls of collapsed lava tubes and crater rims where Daedalus would not dare to tread.

Second, the scout is to be used as a navigation and scientific aid to Daedalus. Instead of expending the time and energy for Daedalus to explore several potential routes along its local path, we can send the scout up ahead to return sparse information about the upcoming terrain. Scientifically, we can accomplish image gathering more quickly with the scout. For example, if we found a spire or small crater of which we wished to gather images, instead of moving the bulky and power consuming Daedalus frame around and rotating to get images from every view, we could send the scout out to scurry around and zap back images as quickly as possible.

The mechanism for the scout would have six independent legs as with Hannibal; this means six small motors attached to six rigid members. The legs would be made longer and the bases of the legs places somewhat further apart in order to achieve a longer stride than Hannibal. This modification would cost us some power due to increased mass, but would allieviate one of the speed bottlenecks. The main aim for the scout design would be for greater mobility and obstacle overcoming abilities than are evidenced in Hannibal. Other issues in the scout mechanism would be mass conservation and energy conservation which would be primary concerns for our mission while it was only incidental to Brooks' situated, embodied, but, unfortunately, goal-less robot. Note that we are concerned with energy and not power (in a sense) because the scout would only recharge when it is in contact with Daedalus.

Sensing would be include a video camera, any force sensors and encoders which would be found on Hannibal, and a spot laser rangefinder. The video camera would be used for navigation and for relaying scientific information back to Daedalus. Force sensors and encoders would be used internally for motion control as with the MIT Mobots. The spot laser rangefinder would be used to help map caves and to help in positioning the Scout or Daedalus in relation to obstacles and goals. This would also be used to measure the size of interesting lunar terrain features (ridges, spires, boulders, gas stations...). This is not a major energy concern since it would only be used sporadically and is a spot sensor. A thermometer would be a cheap and simple additional sensor which would be useful in both navigation (if we wanted to avoid certain temperature ranges) and scientific exploration. All of the sensors would be front mounted. Panning and tilting would be accomplished only with the body.

Communication with Daedalus during separation would be with some sort of low-bandwidth 2-way communications. from Daedalus, we would need general motion directions. Daedalus would be responsible for "long-range" path planning for the scout. In some way, Daedalus would create a gradient field for the scout, where each point was associated with a request "tend to head in THIS direction." This communications path would also be used to relay commands like "gather images now."

Coming from the scout to Daedalus would be sensor information. For scientific purposes largely, we can relay high-res images slowly over the low bandwidth communiocations by merely

sitting and taking our time. We can relay low-res images quickly enough for use in navigation and scout planning, both of which would be handled largely on board Daedalus (we could possibly use neural networks for local navigation with all computing on Daedalus). This communications pathway would also be used to send other sensor (thermometer, etc.) readings to Daedalus at a leisurely rate.

During phases when the scout is physically connected to and being carried by Daedalus, communication would be over a high speed direct connection. This is so that the sensors on Daedalus can be used in real time or Daedalus and so that we could store images gathered during separation in the scout and transmit them back to Daedalus rapidly upon reconnection if we so desired. Reprogramming of the scout would occur only during this phase.

Connection with Daedalus is on the underside. The front (sensor mount) of the scout would face forwards in a fixed direction on Daedalus so the sensors would be of use to the framewalker. Detachment would be accomplished in two phases. First, Daedalus lowers until the scout is almost touching the planetary surface, and then the scout is released and follows the directional gradients given by Daedalus. Reconnection is achieved by having Daedalus call back the scout and then squatting down. The scout then stands up as high as possible and clamps into place with a yet to be designed mechanism.

Why have we chosen this scout design? The mechanism is simply because, the technology is not completely unknown unlike with a snake, we have greater rough terrainability than wheeled robot, and greater sensing capabilities than the ballistic robots. In the far future, a snake robot might be more optimal because of the extreme terrainability, but given the state of the art, this is quite impractical. the sensors were chosen. the sensors are chosen to compliment those on Daedalus and to provide a great flexibility in the scout's mission. With this scout, we can improve overall mission performance both for the old mission of searching for lava tubes and for the new more general lunar exploration mission. Much remains to be fleshed out, but we believe the general design to be sound.

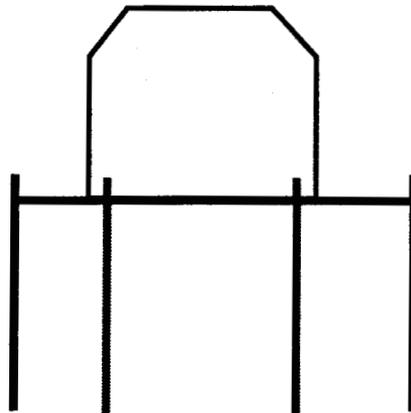


Figure 2.5-1 Scout position on Daedalus. Note sensors are forward



Figure 2.5-2 Scout seen from side

3 Daedalus mechanical design

The requirements in Chapter 1 and the configuration discussion in Chapter 2 are required to create a rover design. However, the converse is also true. One can speak of any imaginable configuration, but unless some design issues have been carefully considered, the configuration may be unrealizable. The Ambler presents an interesting example: while being configured, it was known that slip-rings would be required to enable the circulating gait. However, the cost and impact of the slip-rings on the overall system design was not fully appreciated at the outset.

This chapter addresses many of the design details that are needed to validate the configuration presented in Chapter 2. This chapter is divided into three main sections: The first is the design of the vertical actuators, the second is the design of the horizontal actuator and the y-frame, and the third is a discussion of the body design.

3.1 Vertical actuator

The vertical actuator is comprised of the following components: the leg tube, the mechanical plate, the motor/gearbox, the foot, electrical wiring and connectors, and several sensors. Figure 3.1-1 shows the entire actuator assembly and a close-up of the section around the mechanical plate. Certain details have been eliminated from these pictures for clarity.

3.1.1 Assembly requirements

3.1.1.1 Member sizing

The size of the robot's legs is determined from the equations that describe the most probable failure mode. Since the legs are long, slender rods, so the most likely failure mode is buckling. The equation for column buckling is given by

$$P_{cr} = \frac{n\pi^2 EI}{l^2} \quad 3.1-1$$

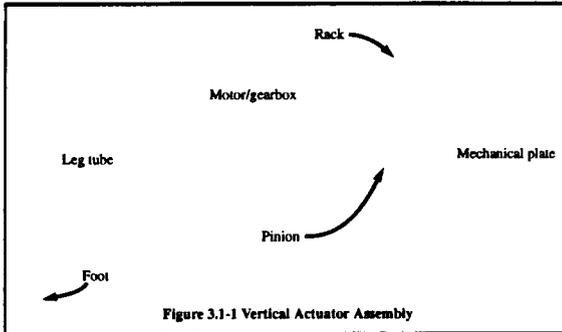


Figure 3.1-1 Vertical Actuator Assembly

29

where P_{cr} is the critical load, n is a constant that depends on the conditions at the ends of the column, E is the modulus of elasticity for the material used, I is the moment of inertia of the beam and l is the length of the beam. Since each leg of the robot is to be capable of supporting the entire weight of the robot, and since the size of the robot is known, the only unknown in Equation 3.1-1 is I . Rewriting this equation yields

$$I = \frac{Nmg l^2}{n\pi^2 E} \quad 3.1-2$$

where m is the mass of the robot, g is the acceleration of gravity and N is a factor of safety. Assuming fixed-rounded end conditions ($n=1.2$), that each leg can support the entire weight of the robot ($m=200\text{kg}$), steel leg 1.25 m long, and a factor of safety of 2, the required moment of inertia is found to be

$$I = 3.92 \times 10^{-8} \text{ m}^4 \quad 3.1-3$$

Although the Daedalus is designed to walk with all legs being vertical, this may not always be the case. To supplement Equation 3.1-2, the equation for maximum stress in a beam due to an applied moment is also used,

$$I/c = \frac{M}{\sigma_{max}} = \frac{mgl \tan \theta}{\sigma_{max}} \quad 3.1-4$$

where c is the radius of the leg, θ is the angle from the vertical of the leg and σ_{max} is the tensile strength of the material. To determine an appropriate angle, the energy stability of the rover will be used, since the leg will not be supporting any load if the tip-over angle is exceeded, Section 2.5. The greatest tip-over angle occurs when the supporting legs are fully retracted, thus no loading is experienced by those legs. The greatest loading occurs when the legs are fully extended, however, the tip-over angle is a minimum at that point. To resolve this problem, an equation relating extended leg length to load is developed. This equation neglects single leg that would tend to prevent the tip-over from occurring. First, the center of gravity is defined, Figure 2.3-3. The CG is then used to define the tangent of the tipover angle, $\tan \theta$ as a function of the extended leg length, l . Not surprisingly, the maximum moment occurs at full leg extension. The angle is approximately 15 degrees, and the moment is 655 Nm, leading to

$$I/c = 1.00 \times 10^{-6} \text{ m}^3 \quad 3.1-5$$

3.1.1.2 Power train sizing

To determine the sizes for the motor and gear train, certain assumptions about the robot's nominal walking cycle are needed, including its nominal speed and a motion profile. For the Daedalus, a nominal speed of 10 m/min is desired. The joint kinematics profile for the Daedalus is a trapezoidal velocity profile, Figure 3.1-2. To simplify the analysis, the rover-terrain interactions will be ignored, since they are discrete and in compression, not shear.

For each motion phase, see Section 2.4, the distance x to be traversed is known, and the phase time T and acceleration time, t , need to be determined. The summation of the six phase time that constitute a single walking cycle is given by the nominal speed of the robot. The goal is to determine phase times and acceleration times for each of the six phases that yield a minimum maximum power requirement. Other possible criteria are possible, such as minimum average power or minimum total energy, but this was chosen since the power sources typically used for space applications are power limited, not energy limited.

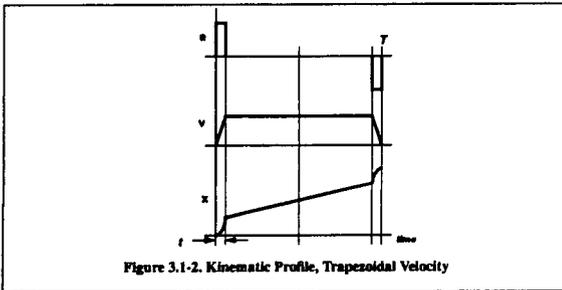


Figure 3.1-2. Kinematic Profile, Trapezoidal Velocity

There are three primary contributors to power consumption of a moving body, inertial power, gravitational power and frictional power. There are other possible power consumers, such as aerodynamic drag, but they will be ignored. A non-regenerative system is assumed, thus the inertial and gravitational energies are not conservative.

The governing equations to determine the power needed to accelerate a body are

$$P = Fv \quad F = ma \quad 3.1-6$$

From Figure 3.1-2 we can write the following equation for the position

$$x = at^2 + v(T-t) \quad 3.1-7$$

where x is the joint displacement and is known. Note that this formulation is not completely general since the two acceleration times are taken to be the same. For motion parallel to the gravity vector, the power optimal trajectory would have different acceleration times. From Figure 3.1-2 and Equation 3.1-7, it is clear that the maximum power occurs at $t = t$. At this time, the velocity is given by

$$v = at \quad 3.1-8$$

so substituting Equations 3.1-7 and 3.1-8 into Equation 3.1-6 yields

$$P_{inert} = mav = ma^2 t = m \left(\frac{x^2}{t(T-t)^2} \right) \quad 3.1-9$$

Gravitational power is given by

$$P_{grav} = mgv = mgat = mg \frac{x}{T-t} \quad 3.1-10$$

and frictional power losses are given by

$$P_{fric} = \mu Nv = \mu kmgat = \mu kmg \frac{x}{T-t} \quad 3.1-11$$

where N is the normal force and k is the fraction of the weight that comprises the normal force. This fraction is used to determine friction when the motion is parallel to the gravity vector.

31

Maximum power consumption is given by the summation of Equations 3.1-9, 3.1-10 and 3.1-11

$$P_i = m_i \left(\frac{x_i^2}{t_i(T_i-t_i)^2} \right) + m_i g \frac{x_i}{T_i-t_i} + \mu k m_i g \frac{x_i}{T_i-t_i} \quad 3.1-12$$

where the subscript i has been added to distinguish the six motion phases. To minimize the power expended, the length of time spent in each of the phases, T_i , and the acceleration times, t_i , are calculated such that the power expended in each phase is a constant, $P_i = P$.

Table 3.1-1 lists the need for including gravitational power, the mass being moved in each phase, m_i , and how far it moves in each of the phases, x_i . The masses listed are the mass of a set of three legs, m_l , the mass of the y-frame with its three legs, m_y , and the mass of the body, m_b . The displacement are given as a fraction of the rover nominal dimension, L , which is a fraction, 0.95, of the diameter of the launch vehicle payload fairing. The vertical excursions are based on the probable size of obstacles along the path and the horizontal excursions are based on the size of the horizontal actuator mechanism. It is clear that phases 1 and 4, and phases 3 and 6 are identical since these phases represent the same motions but with a different set of legs. What is not as clear is that phases 1 and 4 are different from phases 3 and 6. The maximum power expenditure in phases 1 and 4 occurs while the leg is being accelerated to its nominal velocity, and this power is given by $P = P_{inert} + P_{grav} + P_{fric}$. In phases 3 and 6, the maximum power expenditure occurs when the leg is being decelerated to rest, and this power is given by $P = P_{inert} + P_{grav} - P_{fric}$.

	Gravity	Mass	Displacement
Phase 1	x	m_l	0.25 L
Phase 2		m_y	0.90 L
Phase 3	x	m_l	0.25 L
Phase 4	x	m_l	0.25 L
Phase 5		m_b	0.90 L
Phase 6	x	m_l	0.25 L

Table 3.1-1 Phase Power Parameters

From the nominal speed requirements, Equation 3.1-12 and Table 3.1-1, a set of five equations and four inequalities in nine unknowns can be developed:

$$P = f_i(T_i, t_i) \quad i = 1, 2, 3, 5 \quad 3.1-13$$

$$T = \sum T_i \quad i = 1, \dots, 6$$

$$0 \leq t_i \leq T_i \quad i = 1, 2, 3, 5$$

This set of relations is particularly difficult to solve because the bounds of the inequalities are a function of the solution to the problem. Instead of solving this set of relations, a simple relationship between T_i and t_i will be developed, thus reducing Equation 3.1-13 to five equations in five unknowns. Minimizing P in Equation 3.1-13 represents the optimal solution to the problem, so any

32

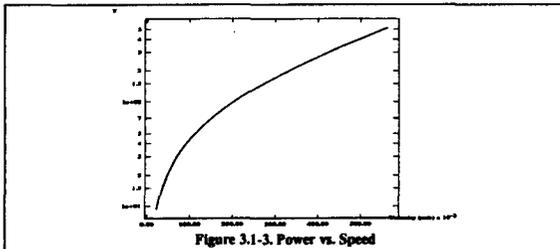


Figure 3.1-3. Power vs. Speed

Other solution to the problem, such as the one proposed, i.e. developing a simple relationship between T_i and t_i , will require more power. For design purposes, this is acceptable since designing for the sub-optimal solution will guarantee satisfying the optimal conditions.

The equation $\frac{\partial P}{\partial t} = 0$ is solved to find a relationship between T_i and t_i . Substituting Equation 3.1-9 for and solving for t yields $t = T/3$. The power to overcome gravity is a function of velocity, so the minimum power required would require $t = 0$, thus resulting in the minimum average velocity. However, the inertial power for $t = 0$ is infinite, so some other value must be selected. Since the minimum power lies within $0 \leq t \leq T/3$, the value $t = T/3$ will be chosen. Substituting this into Equation 3.1-12 yields

$$P_i = \frac{9m_i x_i^2}{2T_i^2} + \frac{3m_i g x_i}{2T_i} \pm \frac{3\mu k_i m_i g x_i}{2T_i} \quad 3.1-14$$

Using Equations 3.1-13 and 3.1-14 a set of five equations in five unknowns can be developed.

$$\frac{a_{i1}}{T_i} + \frac{a_{i2}}{T_i} - P = 0, \quad i = 1, 2, 3, 5 \quad 3.1-15$$

$$2T_1 + T_2 + 2T_3 + T_5 = T$$

where the a_{ij} are a shortened notation for the terms in Equation 3.1-14. To develop a plot of speed versus power, several Daedalus properties must be assumed. Based on the ongoing design work, the following values are chosen: $m_1 = 15$ kg, $m_2 = 25$ kg, $m_3 = 175$ kg and $L = 1.25$ m. Using these values, a power versus speed plot, Figure 3.1-3, is generated.

It should be noted that the plot does not properly represent the real system at the higher speeds because of the increased impact of unaccounted for, non-linear affects.

To design the components of the drive assemblies, knowing the power alone is not sufficient. Instead, the force and velocity profiles of the motions are required. To achieve the given a desired nominal speed for the Daedalus of 10 m/min, the Daedalus must complete 4.44 cycles/min, or

33

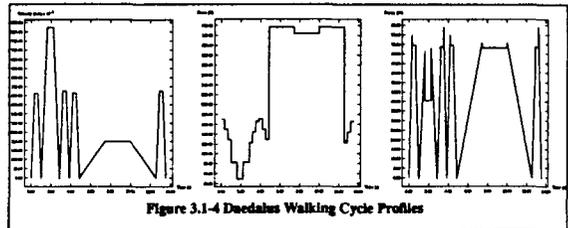


Figure 3.1-4 Daedalus Walking Cycle Profiles

equivalently, 13.3 sec/cycle. Figure 3.1-4 are plots showing the velocity, force and power for one complete Daedalus walking cycle.

Using the power balancing technique, the maximum power expended for the Daedalus to move at 10 m/min is 78 W. The phase times are 1.01 s, 1.84 s, 0.99 s and 7.66 s for phases one (and four), two, three (and six) and five.

It is important to note that these figures do not show the power expenditure for a body lift maneuver. Since this action should only happen occasionally, Section 2.3.3, the actuators will be sized for the power consumption shown here, and geared to provide the higher, required torque, Section 3.1.2.2. For example, to lift the body 0.25 meters with 78 W of power using the motion profile in Figure 3.1-2, requires approximately 8.3 s (1700 N). Although this is not an inordinate amount of time, should frequent body lifts be required, the overall speed of the Daedalus will be significantly diminished.

To improve the accuracy of the power estimate, a simplified model of foot-terrain interaction could be incorporated. This is very difficult, however, because of the complexities of the interaction, the wide range of soil types that can be encountered, etc. It is clear, however, that the work used to compress the terrain under the feet will consume power in excess of that calculated.

3.1.2 Component design

3.1.2.1 Member design

There exist an infinite number of solutions to Equations 3.1-3 and 3.1-5, but the desired solution is the one that is the lightest. Theoretically, a tube with a very large diameter and very thin wall would be the best solution. However, such tubes do not exist and can not be easily manufactured, Section 3.1.3. Instead, a standard tubing will be used for preliminary design purposes. Such a tube is a 1.5 inch OD tube with a 0.049 inch thick wall.

The design of the Daedalus call for linear actuators and the tube described above serves as both the load bearing member and the linear bearing element. (Contrast this to the Ambler which has large aluminum channels as the load bearing members and steel rails as the linear bearing elements.) Like the Ambler, a separate gear rack is mounted on the leg to provide a means of power transmission. This design reduces the overall weight of the system, but also poses three challenges

- Limiting the rotation of the leg to prevent transverse loading on the motor shaft and cable wrap-up problems
- Providing a hardened steel surface for the bearings
- Keeping the bearings free of contaminants and lubricated.

34

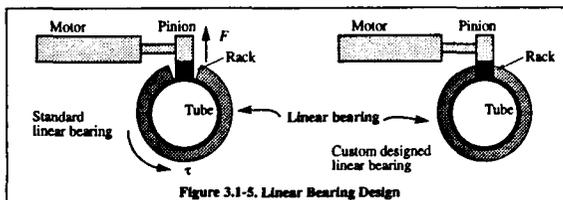


Figure 3.1-5. Linear Bearing Design

The use of traditional open, linear ball-bearing would result in a system that is subject to all of these difficulties. To alleviate the second and third problems, solid linear bearings with Teflon based bearing surfaces used as the anti-friction element are used. These bearings can be used on softer surfaces than ball bearings, albeit with decreased performance, and they are specifically designed to be used in harsh environments where they can not be regularly maintained.

The solution to the first problem requires going beyond standard components. The opening in the standard linear bearing, Figure 3.1-5, does not help alleviate the problem of rotation. If such a bearing were used, some additional hardware would be required to prevent the rotation. Consider instead the custom designed linear bearing, Figure 3.1-5. By making the opening and the sides of the rack into bearing surfaces, rotation of the tube is prevented with a very small mass addition. To prevent the tubes from opening under load, small steel straps, which double as limit switch mounts, will be used.

3.1.2.2 Power train design

There are three issues that need to be addressed: the type and size of the motors, the size of the rack and pinion drive and the design of the speed reducers.

Motor selection encompasses two topics: the type of motor to use and the size of the motor. To achieve the highest motor output power to mass ratio, brushed DC servo motors are chosen. Although brushless motors have a higher power to mass ratio, for small motors, when the mass of the additional commutation electronics are included, the brushed motors' superior power to mass ratio becomes apparent. From Figure 3.1-4, the motor should be capable of delivering 80 W of output power. In addition, the motors must be highly reliable and proven in rugged applications. Motors used by the aerospace community that use cobalt-samarium magnets meet both of these requirements.

[Author's note: English units are used in places because American component suppliers and machine shops still use this quaint system of measurements.] The rack and pinion must be sized to withstand the power and applied loads. Factors taken into consideration for the design of the rack and pinion include fracture due to excessive bending loads, fatigue failure due to repeated loading, surface abrasion due to the contact pressure and thermal loading due to the sliding contact at the interface. To perform a gear analysis, a pitch (teeth/inch) must first be selected. Choosing a pitch of 24 teeth/inch, an analysis of the gear set, based on Lewis' equation (Shigley 77), the gear face is found to be 0.25 inch.

Having designed the rack and pinion, the speed reducer can be designed. To design the speed reducer, the velocity and force plots from Figure 3.1-4 must first be converted into rotational velocity and torque. To prevent tooth under-cutting, the smallest pinion gear with 24 pitch is 0.75 inches

35

in diameter. Using this size pinion gear, the leg velocity (in m/s) is converted to rpm by multiplying by 1003 and force (N) is converted to ounce-inches by multiplying by 1.67. From Figure 3.1-4, the highest speed of the vertical actuator is 0.5 m/s (500 rpm) and the greatest torque required is 180 N (300 oz-in). To simplify the system design, the motor will be sized to handle the maximum load at maximum speed, therefore, it must be capable of handling 90 W, not 80 W.

Motors exist that have no load speeds of 20,000 rpm and stall torques of 10 oz-in. Using a motor of this type and gearing it 40:1 will provide the necessary range of velocity and torque for the nominal walking cycle. However, this gearing will not suffice to lift the body, which requires 1700 N (2839 oz-in).

This is a difficult problem to solve gracefully because of the two different motion profiles: high torque, low speed and low torque, high speed. One solution is to add a clutch that selectively engages different gearboxes for the different motion profiles. A combined gearbox/clutch system could be built to satisfy the requirements, but it would be fairly massive, would require extra electrical wiring and may not be very reliable. A better solution is to use a single epicyclic gearbox with two gear trains and two motors. This design has the advantage that the gearbox mass is reduced and the system is more reliable since the failure of either gear train does not disable the other and each axis has two actuators.

3.1.2.3 Mounting plate design

Thus far, several components of the vertical drive actuator have been discussed: the leg tube (which is both the load carrying member and the linear bearing element, the motor/gearbox, the rack and pinion drive and some electrical wiring. These must all be combined to create the actuator assembly. In addition, since this is a test robot, it is possible that a leg may fail and need to be replaced, thus ease of maintenance is an important issue. (The Ambler was not designed with this consideration and the process of removing and replacing a leg took several days, although some of this is due to its larger size.) To minimize weight and to simplify assembly procedures, the mounting plate is designed to be the bearing surface, motor mount, attachment plate and electrical cable mount point. To remove a leg from the Daedalus will require removing four bolts and the electrical connection. To simplify rover maintenance, the electrical connection will be made simultaneously with the mechanical connection. The one part of connector is mounted to the mechanical plate, and the other to the leg mount plate, thus assuring positive alignment and a cleaner design.

3.1.2.4 Foot design

Since the feet of the Daedalus do not rotate, the foot design can be quite simple. However, a sensor is required in the sole of the foot to detect contact with the ground. The simplest and most robust way to do this is with a piezo-electric crystal. Other alternatives considered include contact switches (moving mechanical parts in frequent contact with the ground may fail), traditional force/torque sensors (too big, heavy and complex) and proximity sensors (the data is not always reliable as it depends on material properties). When strained, a piezo-electric crystal releases electrical energy. This energy is proportional to the force applied. Thus, by measuring this energy, the foot contact force is known. While it is true that these devices are not very accurate and that the data may be noisy, it suffices for this particular application.

To incorporate a piezo-electric crystal, the foot must be comprised of two parts: one that is rigidly attached to the leg tube and one that is free. Since the crystal is electrically conductive, the foot materials must be non-conductive. An appropriate material for this application is some type of plastic.

36

3.1.3 Component manufacturing

3.1.3.1 Member manufacturing

The original idea for manufacturing the leg tube was to use a piece of drawn stainless tubing. However, there are two problems: first, this tubing is quite soft, Rockwell B 80 and second the dimensional tolerance is ± 0.010 in. To overcome these problems, a piece of tubing, with an appropriate inner diameter and oversized outer diameter, will be hardened, then ground to the final dimension. Herein lies another problem, heat treatment tends to distort the object being treated, especially large, thin-walled objects. To overcome this problem, a thick walled tube will be heat treated then ground. Stainless steels are not amenable to hardening, so an appropriate alloy is required. Discussions with engineers at heat treatment companies led to the selection of alloy 4140, which can be hardened to Rockwell C 40. To protect the tubes from the elements and to provide a better bearing surface, the tubes will be treated with the Armalloy process as a final step. In addition, the tubes will have a shallow groove cut in them for rack alignment.

The suppliers and estimated costs for manufacturing the leg tubes are (Note: these estimates are for eleven tubes, one "scrap" for the manufacturers to work with, eight for the AEX with two completed, assembled legs, and two spare tubes):

- Raw tubing material: 11 pieces, 2-1/4 inch x 3/8 inch wall tubing, 64 inches long, net weight 441 pounds. Although this tubing is significantly larger than required, it is the only suitable material that is commonly stocked. The larger size costs us extra money to purchase, to ship and to grind.

Joseph T. Ryerson, Inc
Box 1919
Pittsburgh, PA, 15230
Contact: William T. Paul, (412) 276-5400 x0220
Quote number: Q11FA77921022
Approximate cost: \$1050 shipped to Pittsburgh or Cleveland

- Heat treatment: Heat treatment processes can yield one of two types of finishes, case hardened, which only treats the surface in a small region beneath it and full hardened, which treats the entire part. Since a large amount of material is to be removed, case hardening is inappropriate.

Lindberg Heat Treatment
Solon, OH
Contact: Tony Ross (216) 248-4000
Cost estimate: \$155

Process: Heat, quench, temper, check straightness, straighten if necessary, stress relieve
Tolerance: 0.25 inch over the length
Note: A 1/2 in diameter hole, one in from the end is required for the hanger

- Grinding: Because of the large amount of material that has to be removed, a two step procedure will be used. The first step will be to turn the tubes on a lathe and the second will be the finish grind.

Boston Centerless
Boston, MA
Contact: Jim Taylor, (617) 321-4000
Tolerance: 0.0005 overall, 16 micro inch surface finish, will straighten as required
Cost estimate: \$2500

- Armalloy: The Armalloy process is a chromium finish that improves the underlying materials' wear, corrosion processes and surface hardness. Unlike other chroming processes,

Armalloy is a very thin layer, that is, it does not affect the finish dimension of the part being treated, and it is guaranteed against chipping, peeling, etc.

Armalloy of Western Pennsylvania
1231 Rodi Road
Turtle Creek, PA, 15145
Contact: Greg Ehalt, (412) 823-1030

3.1.3.2 Power train manufacturing

[Note: The design of the power train is being handled entirely by Jim Hart.]

- Motor/gearbox/encoding: The package is comprised of two samarium-cobalt, brushed DC motors driving an epicyclic gearbox to provide both high speed/low torque and low speed/high torque modes. Each motor has a toggle-on/toggle-off shaft brake. Relative position encoding is provided by a rotary variable transformer (RVT). Absolute position encoding is provided by a precision encoder.

Sussex Gear Company
22 Glenbrook Road
Ogdensburg, NJ, 07439
Contact: Jim Hart, (201) 827-2439
Cost estimate: \$2500 per actuator (also see)

- Size: Approximately 2-3/8 inches x 3-3/8 inches x 6 inches, weight < 2.5 pounds
- Rack and pinion: The rack will be supplied by Sussex Gear. It is still unknown if it will be a single piece or multiple pieces. Attaching the rack to the tube is a challenge. The rack is to be bolted from behind, with tapped holes in the rack. Paul Viola (see below) has developed a concept that should enable this assembly.

3.1.3.3 Mechanical plate manufacturing

The mechanical plate serves a multitude of purposes and requires some rather tight tolerances. To minimize the weight, the part will be formed by welding a section of aluminum tubing to a piece of aluminum angle. This part will then be stress relieved, then machined. To ensure dimensional accuracy, all tolerance cuts must be done with the part bolted down.

- Mechanical plate:
MRS
Contact: Paul Viola
Cost estimate: \$1000 (quantity 8)

3.1.3.4 Foot manufacturing

The foot is manufactured from Torlon 5030, a glass reinforced amide-imide polymer and a piezo-electric ceramic material. The foot will be manufactured by MRS (see above).

- Piezo electric material: Lead-zirconate-titanate ring, 3/4 inch OD x 1/4 inch ID x 1/4 inch thick

American Piezo Ceramics
Duck Run Road, P.O. Box 180
Mackeyville, PA, 17750
Cost: \$150 (quantity 8)

3.1.3.5 Other vertical actuator components

In addition to the parts specifically listed above, several other small parts are required, including the limit switch mounts, the e-chain holders, the connector mount plate and the leg-tops. These parts will all be provided by MRS.

37

38

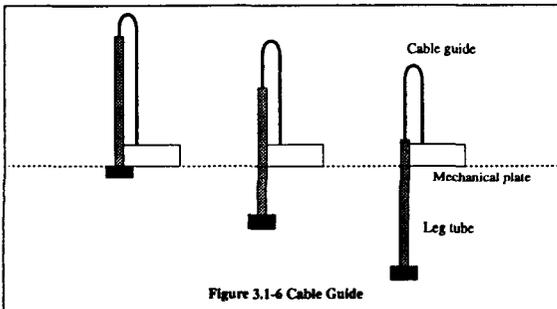


Figure 3.1-6 Cable Guide

3.1.4 Sensors

The vertical actuator makes use of three types of sensors: positioning sensors, force sensors and over travel sensors. The first two have been previously discussed in Sections 3.1.3.2 and 3.1.3.4. Over travel sensors are required to ensure that the leg does not hit the mechanical plate. This could potentially damage the leg and possibly the absolute encoder. Two technologies are currently being considered for implementing the over travel limit, LED and inductive proximity switches. Plunger type mechanical switches are not being considered, except as a last resort, because they have moving parts, and are therefore less reliable than the other technologies considered. A sample of each has been obtained and tested, but a final determination of the suitability of these devices requires the completed actuator mechanical assembly. The sensor is mounted to the mechanical plate by a steel strap, the design of which is easily altered to fit either sensor.

3.1.5 Electrical cabling

One of the goals of the vertical actuator design is to keep the number of wires to a minimum. By using brushed motors, instead of brushless, an RVT, instead of an encoder, and the piezo-electric force sensor, instead of a force/torque sensor, the total number of wire required by the leg is reduced to 20.

To guide the wires from the moving part of the leg to the mechanical plate, the cables must be managed in some fashion. The vertical actuator will make use of plastic e-chains to serve as cable guides. A problem with plastic e-chains is that should a link break, the entire must be removed to replace the link. This problem can be minimized by using a connector on one end of the cable. Figure 3.1-6 shows the scheme for guiding the cables.

- Cable guide: These plastic e-chains will work at the required speeds.

Igus
P.O. Box 14349
East Providence, RI, 02914
Contact: (800) 521-2747
Part number: 06-06-018 (colors available) with brackets 060-10-1

39

- Connectors: To implement the quick connect concept, a pair of panel/panel mount is required. The connector must be capable of handling currents of up to 5 amps, reliable, keyed, corrosion resistant and weather-proof. To provide for future expansion, extra pins should be available, although a large connector is not desired.

Lemo USA
P.O. Box 101
Hickory, PA, 15340
Contact: Gary Farmer, (412) 356-2305
Part number: FAA/ERA 4S 24 C L A/L

3.1.6 Control

- Servo amps - Elmo servo amps, model number SSA-8/80, rated at 3 times the required power rating will be used. These are very efficient and in a very small package. The cost about \$350 each.

3.2 Horizontal actuator

3.2.1 Requirements

To size the y-frame, it is not the failure of the frame that is a primary concern, but rather its rigidity. To size this frame, the maximum deflection of the frame is expressed as a fraction of the total leg height. For purposes of this analysis, the frame can be treated as a simply supported beam with the load occurring off center. The formula for the maximum deflection is

$$y = \frac{A^2 B^2 F}{3EI} \quad 3.2-1$$

where F is the load, A is the position of the load along the beam and B is the length of the beam less A . Again, the only unknown is the moment of inertia, so Equation 3.2-1 is rewritten as

$$I = \frac{NA^2 B^2 mg}{3YE} \quad 3.2-2$$

where Y is the fraction of I of permissible deflection. Values for A and B are obtained from Y , Y is chosen to be 0.01 and the safety factor is 2. Solving Equation 3.2-2 for I yields

$$I = 1.81 \times 10^{-8} \text{ m}^4 \quad 3.2-3$$

The lightest, available tube that can meet this requirement is a 42 mm with a 0.7 mm thick wall, with a linear density of 0.402 kg/m. The mass of the frame is 1.023 kg.

The masses given for the legs and frame do not include motors, gear rack, structural support elements, etc. However, these values will be used to provide a first estimate of the power requirements, see Section 2.3.3. Once the power requirements have been determined, the masses of these elements can be calculated, then final power figures can be obtained.

40

3.2.2 Component design

3.2.3 Component manufacturing

3.2.4 Sensors

3.2.5 Electrical cabling

3.2.6 Control

3.3 Body and sensor mast

The design of the chassis is particularly challenging because of the wide variety of instruments that must be accommodated. The chassis will be based on a framework of small diameter, titanium tubing. These tubes will have flats and flanges welded on to accept bolts for equipment. To protect the equipment, the framework will be covered with space-type solar reflective material. Although this will not provide the same type of protection that a rigid panel would, it is significantly lighter. Furthermore, since the use of a large number of robots is expected, the loss of any single robot will not kill the mission.

3.4 Foot Design

3.4.1 Introduction

The motivation behind this section is to investigate design issues related to Daedalus's feet in order to insure that Daedalus will be able to safely walk over rugged terrain. This includes not only structural and mechanical issues, but perception issues as well. In order to walk safely, the foot must be designed for maximum stability of footfalls, and given sufficient sensing capabilities to find areas of good stability (either directly through perception, or via additional sensors).

The design of the foot can be broken into three major areas of investigation: 1) Foot sensing, 2) Foot Shape, and 3) Structural design. Foot sensing addresses issues concerning the choice of sensor suites to be imbedded within or in the immediate vicinity of the foot in order to provide feedback for control and planning of leg/foot placements. Foot shape addresses the shape, size, and mechanical characteristics of the foot, and investigates the differences in foot-terrain interaction of various foot configurations. Structural design will address the forces and stresses put on the foot in order to ensure that the design will stand up to worse case loading and extended environmental interaction. In the following sections, a detailed evaluation of options available and the thought process leading to the final design decisions will be presented.

3.4.2 Past Experience

We begin the design process by examining the foot design on Ambler and Dante.

3.4.2.1 Ambler

Ambler had large, flat pancake feet. The shape and size of the foot were primarily driven by the goal to reduce ground contact pressure, maximize traction, and to be able to compact soil when walking on soft slopes. Walking relied on terrain maps built from the perception system, as well as information given by a IR³ 6-axis force sensor located on each foot. No additional contact sensors or proximity sensors were used.

41

3.4.2.2 Dante

Dante's foot was 3" in diameter and had a concave tread. The shape of the foot was motivated by



the desire to maximize traction, with the convex shape giving the best traction in sandy soil (NEED PETE NACY'S REFERENCE HERE). The size of the foot was designed to be the same size as the leg (which was already designed and fabricated), in order to eliminate any "hips" that could get caught under rocks, as experienced by Ambler. Imbedded into the foot was a contact sensor, LVDT/Wave Spring single-axis force sensor, and a Capacitance proximity sensor.

Foot Shape: 3" diameter concave tread.

1. Contact sensors were unreliable. Tendency to get stuck in on/off positions. Ended up not being used for the most part. Most likely just a bad implementation/design, and nothing inherent in the general idea of contact sensors.
2. Capacitance sensors never used. Problems with calibration and interpretation of data.
3. Force sensor used as contact sensor with adjustable trip thresholds. Thresholds depended on slope of terrain.
4. Force sensor not as precise/repeatable as desired.
5. Foot shape resulted in "punch-through," especially on snow.

3.4.2.3 General Notes

1. Force sensors were used as fancy contact sensors, for the most part.
2. With both Ambler and Dante (Erebus), force sensors were never used to equalize weight distribution on all legs during normal walking cycles.
3. With Apex which has three only three legs on the ground (ignoring the transition stage), there is no need for active force control to equalize weight distribution (sorry, active force control group).
4. Moon/Earth gravity differences will need to be taken into account with contact sensors on the Apex.

3.4.2.4 Summary & Future Agenda

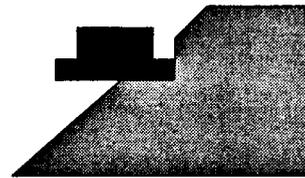
1. Past experience with Ambler and Erebus indicates that simple contact sensing is all that is required for most walking applications.
2. Force sensors have been used as glorified contact sensors. Has the advantage of being able to dynamically set contact thresholds.

3.4.3 Foot Sensing

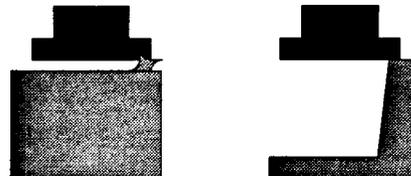
Two conditions are needed to ensuring stable footfalls. First, enough must be known about the environment (and mechanism) to select locations that minimize the possibility of the foot slipping or terrain giving away (a perception problem). Second, the foot must be designed in such a way as to maximize the choice of terrain conditions in which a footfall will be stable (foot mechanical design problem).

Past experience with Ambler and Dante has shown that soft terrain does not pose much of a problem. Issues such as sinkage and traction could be ignored in the types of terrain that both robots

43



The force sensors were used primarily as vertical and horizontal contact sensors during the walking cycle. Only three of the six axis of force were used for any extended walking, and the sensors themselves were found to be very reliable. Experiments were done to determine if force sensor torque information could be used to signal bad footfalls, and this was found to be difficult in general. In the figure below, the left foot placement is fine, the one on the right is undesirable, while the torque information is the same in both cases.



There were minor problems associated with the foot being larger than the leg. The lip of the foot had a tendency to accumulate dirt, and would occasionally catch under rock. In addition, the foot would occasionally slip off of footholds and undergo significant vertical drops, especially when other parts of Ambler were in motion (perturbing the foothold). The reasons for this and prevention methods will be presented in later sections (NEED CROSS REFERENCE HERE).

42

traversed. It is expected that Daedalus will walk across similar or harder terrain, where soil deformation is not a factor in mobility.

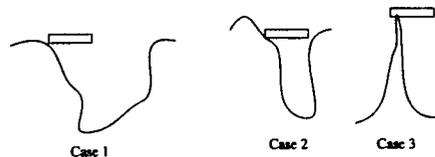
The problem then becomes that of traversing very hard and rugged terrain, where one can expect boulders of all sizes, cracks, crevices, and spikes. The definition of "rugged" itself is ambiguous, and depends entirely upon relative sizes of terrain features and the robot.

We begin with an investigation of what sorts of terrain features would be most problematic for Daedalus. Features very much smaller than Daedalus are not an issue. For example, stepping on a 1cm pebble poses no threats. Features very much larger than Daedalus becomes a navigation and route planning issue. Either a way around/over the obstacle exists, or the path is intraversable. They are not direct threats to Daedalus on a step by step basis (traversing the edge of a large gorge poses the same threat as smaller obstacles, to be described later). Obstacles somewhere between these two extremes pose the most hazard to Daedalus. The hazard, of course, is that Daedalus will lose its footing and tip over.

The problem is to determine the class and characteristics of obstacles that must be dealt with on a step by step basis. This is not as difficult as it initially seems. We are concerned with preventing tipover of Daedalus, and in this regard, the problem simplifies considerably. Tipover is possible if either a foot slips off of its support, or the support breaks down. In either case, the result is the same. Daedalus will experience a torque and its body orientation will shift until something once again reestablishes support on the foot that slipped. The offending foot will experience a verticle drop, and the magnitude of this drop determines the amount of tipover energy imparted to Daedalus.

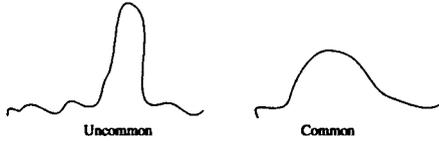
Why the 1cm pebble does not pose a threat to Daedalus becomes clearer. If the pebble crumbles, or if the foot slips off the pebble, the foot will drop only 1cm, and during the short period of time where the forces on Daedalus are not in equilibrium, the angular momentum imparted is not large enough to cause tipover or mechanical damage.

We can pick a nominal vertical drop that could be considered safe for Daedalus (call it V_{nom}). The class of obstacles that must be dealt with is anything that could produce such a drop. Essentially, this is any terrain that has a ledge greater than V_{nom} .



44

In natural terrain, "spikes" are uncommon. The height to width ratio of typical terrain features that



create "ledges" are about 1:1 or smaller. Objects that are taller than they are wider will normally go to a minimum energy state, and fall over.

This gives some insight into perception requirements. We must be able to detect terrain features that can generate a vertical drop that is greater than what is considered safe for Daedalus, and we observe that such features are typically at least as wide as they are tall. This gives a bound on the required resolution of terrain maps. Terrain map resolution must be a minimum of twice the maximum vertical drop tolerable. For example, if the maximum drop tolerable is 10cm, terrain map resolution must be at least 5cm.

3.4.4 Foot Size

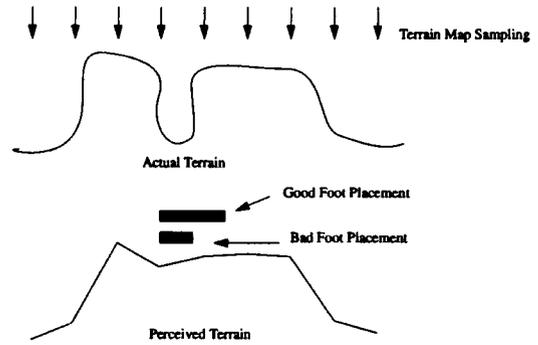
Investigation into how foot size affects sensing requirements reveals a surprising result: Foot size does not affect sensing requirements as long as the foot is larger than the terrain map resolution!



Equally bad placements

In the figure above, if the obstacle height is larger than the acceptable minimum vertical drop of Daedalus, the foot placement is equally bad for both the large foot and the small foot. The terrain feature creating the potentially hazardous foot placement must be sensed, and the requirements for sensing the feature is independent of foot size.

Although spikes are relatively uncommon, it is very common to find holes in natural terrain. This gives the requirement that the foot must be as large as the terrain map resolution.



In the figure above, the terrain map resolution is such that the "hole" between the two "boulders" is missed. A foot size that is smaller than the terrain map resolution has the potential to grip the edge of the hole, then fall into the hole during any perturbation.

3.4.5 Foot Shape

3.4.6 Foot Shape Study

3.4.6.1 Foot Shape Design Criteria

Foot-Terrain interaction Considerations

1. Loading (contact pressure) - prevent excessive soil penetration, soil failure
2. Traction (lateral forces) - sufficient for movement.
3. Stability - prevent foot support failure

Best performance: Maximize foot/ground contact area in direction of force (vertical & horizontal direction of motion).

Mechanical Design Considerations

1. Cost
2. Size
3. Weight
4. Complexity

3.4.6.2 Foot Shape Analysis

Soft Terrain

1. Soil will deform to shape of foot.
2. Shape of foot impacts traction most, with minimal impact on loading and stability.
3. However, traction is not expected to be a problem

Hard Terrain

1. No rigid shape maximizes contact area for all cases.
2. Convex shapes are more stable than concave (forces will tend to center foot on support).



3. Best foot: one that conforms completely to surface, then becomes rigid. You can imagine a fluid turning to a solid, or one of those "nail platforms."
4. Passive ankle: increases contact area in some cases (good), but adversely affects stability in most.



5. Passive ankle gives you opportunity to sense?

3.4.6.3 Conclusion

1. Soft terrain not expected to be a problem, and has very little impact on foot shape.
2. Hard terrain poses the most difficulty.
3. Convex foot shape offers best stability of foot staying on support. Other than this, mechanical design considerations dominate.
4. Foot compliance with terrain is highly desirable.
5. Fixed angle offers more stability than passive ankle, and outweighs benefits of increased contact area given by passive ankle.
6. Additional benefits offered by passive ankle sensing need to be examined.

3.5 Thermal design

In this section, we will address the need of heat removal techniques and its preliminary designs. Although, the first prototype of Daedalus will be tested in terrestrial environment, the designs will take into consideration of various extraterrestrial factors in such a manner that minimal changes and studies are required for the actual prototype to walk on the Moon. The distinction between the thermal system of the earth and the lunar prototype will be indicated wherever necessary. Some fundamental theories on heat transfer are cited to promote understanding at the level of designers. The information from manufacturers and other sources are also presented herein with critical suggestions for achieving practical systems.

3.5.1 Introduction

A thermal system is one of various supporting systems required in every spacecrafts and satellite. Its main function is to control and maintain levels of temperatures of all components within permissible range during their actual operations. While providing important roles, the thermal system, in general, accounts not more than five percents of a spacecraft[1]. Past experiences and studies of these thermal systems can be used as possible references for space robotics.

Daedalus will be the first autonomous robot which operates on the Moon. The major heat input comes from the Sun and the Moon surface. The average heat flux at the orthogonal direction to surface is about 1.358 W/m². The highest surface temperature of the Moon is about 100 °[2]. The radiation from the Moon surface to the Daedalus must then be considered. When combined with heat generated by electronics inside Daedalus, the amount of heat to be removed is substantial. In addition, the vacuum environment makes it difficult to implement some means of heat convection to transfer undesired heat back to environment. Active cooling schemes are usually more efficient than passive ones. However, the total mass and limited power of Daedalus will be critical factors in determining a practical system for thermal control.

In case of the earth model, the thermal system must be designed in such a way that can protect and allow Daedalus to operate properly in thermally fluctuated environment. Under situation like raining during expensive operations, the temperature of Daedalus body changes abruptly. The high rate change of temperature has potential to adversely affect the operating frequencies of some electronics.

We will begin designing of the thermal system by identifying critical objectives in the following section.

3.5.2 Design Objectives

The main objective is to maintain working temperature ranges of all components of Daedalus during a period of mission.

Typical temperature ranges which allow components to function properly are as follows [1].

Components	Typical Temperature Range, C
Electronics	0 to +40
Batteries	5 to +20
Solar Arrays	-100 to +100

Table 3.5-1 Working Temperature Ranges

Components	Typical Temperature Range, C
Structure	-45 to +65
Infrared Detectors	-200 to -80

Table 3.5-1 Working Temperature Ranges

3.5.3 Design Consideration

3.5.3.1 Robust Operation

The system must function robustly in wide ranges and cycles on environmental temperatures. Rates of failure due to thermal fatigue must be minimal during the lunar mission. The mission period of Daedalus is aimed at two lunar days which are about 28 earth-days.

3.5.3.2 Power Consumption

Due to limited power on board, it is necessary that the system consumes less energy. Chapter 9 discusses on the power requirement. The components which require thermal control are also indicated. The total power consumed by these components is about 200 watts. Considerable amount of this input energy will be converted to undesired heat. When combined with radiated heat from the Sun and the Moon's surface, the active thermal system may consume more than 100 watts to control temperatures within limits. The accurate calculation of the energy consumption must take into account of the cooling area, the temperature limits and cooling techniques. Section 3.5.5 will discuss in further detail on the comparison of active and passive systems.

3.5.3.3 Weight

It is critically required that all subsystems within Daedalus have light weight. In case of the thermal system, the ratio of amount of heat removal per weight plays an important role in selecting the feasible techniques.

3.5.3.4 Physical Configuration

Physical configuration of thermal control components will affect the inside space which is severely limited by multi-components of other subsystems. As the priority of space must be provided to those systems, the possible design of the thermal system must wisely benefit from commercially available configurations which satisfy this space constraint.

3.5.3.5 Cost

Low cost in constructing the system is more favorable, although it is not vitally necessary in case of a prototype.

3.5.4 Heat Sources and Consideration on Design Parameters

3.5.4.1 External Heat Sources

The Sun is the original source which radiates heat to Daedalus as well as its surrounding environment. The Stefan-Boltzmann equation describes heat radiation as follow[3].

$$Q_R = A\epsilon\sigma T^4 \quad 3.5-1$$

where Q_R is an amount of radiated heat. σ is the Stefan-Boltzmann, $5.67 \times 10^{-8} \text{ W/(m}^2 \cdot \text{K}^4)$. A is an overall surface which is perpendicular to radiation rays. The absolute temperature, T is in Kelvin. The emissivity, ϵ is the ratio of total energy emitted by a real body to that emitted by a black body at the same temperature and wave length. $\lambda(\epsilon = q_R/q_{B\lambda})$, q is radiated heat/unit area.

49

ϵ of the black body is 1. The same equation is also applicable for heat radiating from the Moon's surface to Daedalus. In this case, the term T^4 is now replaced by $(T_s^4 - T_r^4)$. Subscripts (s) and (r) refer to heat source and receiver respectively.

The cross sectional area, A is a design parameter which indicates how much heat radiates. Daedalus body, as shown in Figure 2.3-1 is symmetrical about the z-axis. Such a physical configuration will allow some variation of 13% of A while the body is rotating in the x-y plane. To minimize numbers of rays falling on Daedalus, panels of solar cells can be design to have another concurrent functionality namely coating off the rays as umbrellas. Note that heat flux to Daedalus is still the same. The umbrella only partitions the heat coming in. Care must be taken on selection materials at the mechanical interface between the umbrellas and Daedalus body. The material selected should have high bending strength and low thermal conductivity.

There is another property for radiation so called radiosity. The radiosity includes reflected heat and emitted heat, as shown in Figure 3.5-1. In general, the selected materials for coating or wrapping wall must allow reflectivity as much as possible. Proper insulators with less thermal conductivity minimize rate of absorbed heat (dq/dt) through the wall. The thickness of this insulator also play role in reducing such a rate (See also section 3.5.6.4). However, we need to concern about space and weight.

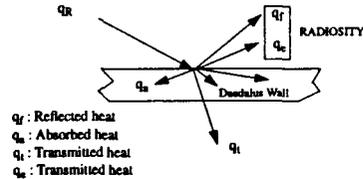


Figure 3.5-1. Radiosity

3.5.4.2 Internal Heat Sources

Today, new electronics and data processing equipments include higher I/O, larger circuit chips and increased circuit density. Greater circuit density means higher performance as well as more heat flux (W/m^2), generated by those equipments. Table 3.5-2 shows the profile of modules, heat flux and technology used for heat removal.

Modules	Heat flux $\times 10^6$ (W/m^2)	Technology
Vacuum tubes	0.2	Air cooled
IBM 3033	0.5	Air cooled
Honeywell DSP-88	0.7	Cold plate cooled
NECLCM	1.3	Cold plate cooled
FACOM M-380	1.7	Air cooled
CDC Cyber 203/205	2.3	Cold plate technology
IBM 3090 TCM	3.4	Cold plate technology

Table 3.5-2 Module Heat Flux

The information from manufactures in term of heat flux of electronics actually used in Daedalus will be gathered in the near future. Such information will help refining our designs of Thermal control systems discussed in section 3.5.6.

Reliability of electronics components is related to operating temperature. Higher operating temperatures will accelerate several failures such as creep, corrosion and electromigration. There have been discussion on temporarily turn on and off some subsystems when necessary, in order to minimize the total power consumption. Care must be taken on maintaining the working temperatures under this special circumstance so that Daedalus electronics will not suffer from the temperature cycle that occurs during power-off and power-on. To reduce temperature spike, we can implement an idea of using paraffin plates[1] to absorb the undesired heat during power-on. Paraffin stores the heat by changing its phase between solid and liquid. During electronics boards are inactive, paraffin cools down and solidified. Temperature of paraffin will be controlled by amount of heat being transferred to the cold end. The high heat capacity of paraffin acts like inertia that allows small changes of temperatures with respect to time.

Temperature also affects the moisture levels inside major IC chips which leads corrosion process. The Memis model[4] shows a relation between temperature and humidity as follow:

$$RH = RH_0 + (RH_{ext} - RH_0) \left(1 - e^{-\frac{t}{\tau}} \right) \quad 3.5-2$$

RH = relative humidity
 RH_0 = initial RH inside the IC package
 RH_{ext} = RH external to the IC package
 t = time
 τ = time constant
 H = activation energy

51

R = gas constant
 T = temperature in Kelvin
 τ = time constant for the diffusion constant = $\tau_0 \times e^{-H/RT}$
 T and RH_{ext} are design parameters which should be controlled properly.

3.5.5 Thermal Control Techniques

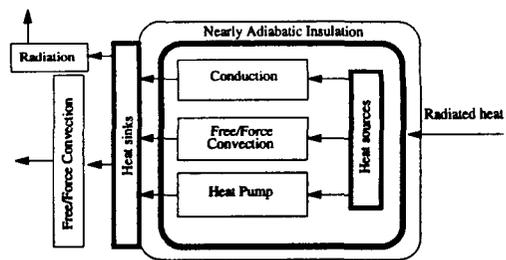


Figure 3.5-2. Thermal Control Techniques

Figure 3.5-2 shows general techniques in rejecting undesired heat. Heat sources refer to subsystems which generate heat inside Daedalus. Heat from these sources can be removed through thermal partitions by conduction, ventilation (free convection), force convection and heat pump. Heat sinks are fin panels and radiators for the earth and lunar prototype respectively. In our case, the most significant criteria is reliability since we cannot afford the system to fail at any circumstance during actual missions. Reliability is generally a function of simplicity in term of less number of components involved in the system. In addition, ratio of heat removal/weight of components is of our concern for practical systems. We will examine mechanics of each technique in this chapter. There are two main types of heat techniques i.e.: passive and active techniques.

3.5.5.1 Passive Systems

By the Fourier law, the rate of heat conduction is a relation among four parameters i.e.: temperature difference (ΔT), thermal conductivity (K), cross sectional area (A) and the distance between a heat source and a heat sink (dx):

$$Q_C = -AK \frac{dT}{dx} \quad 3.5-3$$

Q_C is the rate of conducted heat transfer per cross sectional area. The conduction method is suitable for insulating (very small K) Daedalus while is impractical in case to transfer heat for a

52

long distance, except K is extremely high. Better cooling by conduction based on proper thermal contact/bond between heat sources and sinks[5].

Another form of passive thermal system is introduced the Newton's law of cooling: 3.5-4

$$Q_v = hA\Delta T$$

h is a convective heat transfer coefficient which depends what kind of fluid used to transfer the heat. A is an exposed surface which can be designed by adjusting physical appearances of parts to be cooled. In practice, dust pollutant will lower the rate of heat transfer. Force ventilation with installation of HEPA (High Efficiency Particle Air) filters at the entrance will reduce dust flowing around sensitive electronics. Note that air cooling is impossible in the lunar prototype.

The passive techniques are favorable in term of low cost and reliability. However, they often suffer from the fact that heat removal capacity per weight is much smaller than active systems, to be discussed in section 3.5.5.2. To increase such capacity, the new device so called "Heat Pipe" has been built and tested successfully in many space projects. Heat pipes are available commercially. One type of heat pipes has been installed in "Tessier", a project currently under construction at the Field Robotics Center. The manufacture which we contact is:

Noren Products, Inc.
1010 O' Brien Dr., Menlo Park, CA 94025
(415) 322-9500 Fax (415) 324-1348 Telex 17-1410 NOREN MNPX

The rest of this section will be devoted in describing the mechanics inside heat pipe. We will introduce several types of heat pipes and their functions by purpose of further understanding in this device before pursuing the design of Thermal systems in section 3.5.6. More information can be obtained in [6], [7], and [8].

R.S Gaugler suggested the idea of heat pipe in 1942. Its concept is similar to the thermosyphon. Thermosyphon requires that the condensate to be returned to the evaporator area by gravitational forces whereas heat pipe does not have this limitation. Therefore, heat pipe always functions regardless its orientation. Figure 3.5-3 shows schematic diagrams of both thermosyphon and heat pipe [7]. The wicks in heat pipe is constructed from a few layers of fine gauze which are fixed inside the inside surface. When the heating end(evaporator) receives heat, the liquid inside become vapor and move towards the cooling end. At this end, the vapor will condensate and this gives heat out. Capillary forces return condensate to the evaporator.

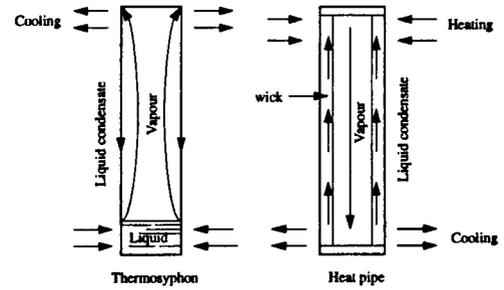


Figure 3.5-3. Thermosyphon and Heat Pipe

We can classify heat pipes by methods of condensate return, as shown in Table 3.5-3.

Method of condensate return	Type
Gravity	Thermosyphon
Capillary force	Standard heat pipe
Centrifugal force	Rotating heat pipe
Electrostatic volume forces	Electrohydrodynamic heat pipe
Magnetic volume forces	Magnetohydrodynamic heat pipe
Osmotic forces	Osmotic heat pipe
Bubble pump	Inverse thermosyphon

Table 3.5-3 Classification of Heat Pipe

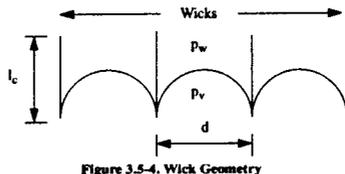


Figure 3.5-4. Wick Geometry

Standard heat pipes benefit from capillary effect at the wicks. In Figure 3.5-5, l_c , d , p_w and p_v are wick depth, wick width, wick pressure and vapor pressure respectively. Taking static equilibrium between surface tension and the pressure difference ($p_w - p_v$), as well as applying two properties law, we obtain the total heat removal:

$$q = \left(\frac{Ad}{8l_c} \right) \left(\frac{L\sigma}{\mu v} \right) \quad 3.5-5$$

where A is project chamber area (not shown in Figure 3.5-5). It can be seen that amount of heat being transfer depends on geometrical parameter and fluid properties. Critical fluid properties are surface stress(σ), latent heat constant (L), viscosity (μ) and specific volume (v). It is because of the latent heat that enhances thermal conductivity of heat pipe. Equation 3.5-5 enable us to do custom design & implementation of heat pipes when necessary. In addition to very high thermal conductivity, heat pipes also have other characteristics:

- (1). A heat pipe can be used as thermal flux transfer by varying the surface of heating and cooling ends.
- (2). The condenser surface of a heat pipe will tend to operate at uniform temperature.
- (3). The gas buffered heat pipe can maintain the heat source temperature at an almost constant level over a wide range of heat input.
- (4). A heat pipe can function as thermal diodes and switches. Thermal diode is crucial for Daedalus as to prevent heat from outside coming back in. Thermal switch is also required to maintain lower bound operating temperature so that electronics will not shock due to power cycles as mentioned in section 3.5.4.2.

In addition to solid heat pipes, there are many physical configurations of heat pipes commercially available in the market[6]. Here, we will briefly mention the ones which have potential to be used in our thermal control system. They are flat and flexible heat pipes. A flat heat pipe has ability to provide a surface with a very high thermal conductance. It is a suitable mounting base for printed circuit boards(PCB). In general, the internal surface of each face is wicked to accommodate either condensation or evaporation. Internal structural supports, for example studs, can be employed to strengthen the unit. Not only a commercial flat heat pipe(AL) weigh less but it also is 2.4 times more efficient when compared with a solid aluminum plates with the same size. If the heat pipe is made of copper, the number is 4.4. With this type, the center temperature of PCB reduces 78% and the virtually isothermal surface can be maintained. Moreover, two PCBs can be mounted on each

side of the heat pipe. It is also possible to place the second layer of heat pipe at 90° to the first one in order to increase heat removal capacity.

Flexible heat pipes are favorable in the situation where the space is severely limited. This heat pipe can also accommodate vibration due to structural dynamics, temperature cycling as well as sudden change in evaporator and/or condenser ends. Flexibility in the forms of bending, expansion/contraction provide some range of motion required. To maintain levels of fluid inside while changing physical configuration, this heat pipe must have an expansion reservoir.

3.5.5.2 Active Systems

The performance of any active thermal systems regardless how complicated it is, can be described in term of heat pump, shown in Figure 3.5-6.

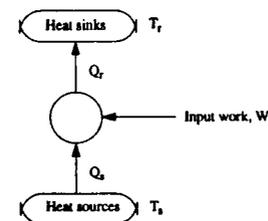


Figure 3.5-5. Heat Pump

The coefficient of performance by the ideal Carnot cycle is defined as follow:

$$COP = \frac{Q_r}{W} = \frac{1}{\left(\frac{T_r}{T_s} \right) - 1} \quad 3.5-6$$

The operating temperature of electronics is about 40° C(315 K). If we design the temperature of a fin panel/radiator outside to be 80° C(355 K), we will obtain COP = 7.87. This means that an active system at least requires 30, 720 KJ to remove 100 watts from electronics boards during its two lunar day-mission. The number will get even higher in practical circumstance where isentropic efficiency and other losses are taken into consideration. Due to limited power Daedalus has on board, it does not look promising to implement the active thermal for the lunar prototype.

3.5.6 Designs

In previous sections, we have briefly discussed an overall picture of thermal control techniques and their design parameters as well as some constraints and limitations. Since detailed

dimensions and physical configuration of subsystems inside Daedalus are not finalized, our preliminary designs will focus on concepts and critical suggestions. This design will be refined when more information is available.

3.5.6.1 Heat Transfer from Electronics Board

References	[6],[8],[9],[13],[14]
Techniques	Lunar model: Passive flat heat pipes Earth model: Passive flat heat pipes, Air Cooling

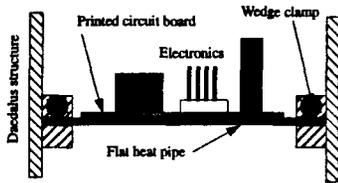


Figure 3.5-6. Flat Heat Pipe

Critical suggestions	<ol style="list-style-type: none"> (1) Strength of commercial heat pipes is for supporting PCB in the vertical direction, not for axial and bending loads due to deformation of Daedalus structure. (2) Avoid shorting circuit by anodizing surfaces of heat pipes. (3) To bond PCBs to heat pipe, temporary glues such as metal-loaded silicon rubber should be used for the earth(experimental) model. For the lunar prototype, silver-loaded epoxy with high K must be used.
Sizes available	2" x 4" to 18" to 24" with thickness of 1/2", 3/8" and 1/4"
Thermal resistance	It can be obtained by experiments. We should consult with manufactures. For example, a standard 4" x 9" heat pipe has thermal resistance of 0.015 ° C/W.
Design equations	We can assume the Fourier law to compute temperature rises.

57

Electrical/thermal analog yield:

$$R = \frac{V}{I} = \frac{\Delta T}{Q} = \frac{\Delta T}{KA} \quad 3.5-7$$

R and equivalent thermal conductivity(K) are provided by manufactures. Equation 3.5-7 then relates the temperature rises with total heat input and dimensions(A) of the heat pipe.

We can compute the thermal storage for a phase-change heat shield like Paraffin, mentioned in Section 3.5.4.2:

$$Q = \rho V \left(\frac{C_p \Delta T + L}{\Delta t} \right) \quad 3.5-8$$

ρ : paraffin density ΔT : Temp. change in paraffin

Q : rate of heat transfer Δt : Change in time

L : latent heat of paraffin C_p : Specific heat of paraffin

Volume (V) and Δt are design parameters in the heat balance equation:

$$Heat_{electronics} = Heat_{stored} + Heat_{removal} \quad 3.5-9$$

Note that heat removal will go through structures which support heat pipes and PCBs. It is still NOT determined yet whether or not these structures will be parts of Daedalus structures. Critical issues are strength, thermal conductance and thermal capacitance. Based on required Q , ΔT and Δt , the thermal conductance and capacitance of the structure materials will determine their dimensions which, in turn will affect strength of the structures.

For air cooling, we recommend that PCBs are cooled locally by mini fans. In stead of using air ducts to convey the ambient air for cooling, we will circulate the air inside Daedalus body. Efficient air circulation can be done by proper selections of air entrances and exits [13],[15]. In dusty environment i.e. deerts, HEPA filters[9] must be installed at the entrance to protect sensitive electronics. Pressure-drop across these fans and air volume flow-rate for cooling will determine a size of the main blower, installed at the exit.

58

3.5.6.2 Heat Transfer from Daedalus structure

References	[6],[9],[14]
Techniques	Passive flexible heat pipes for both prototypes

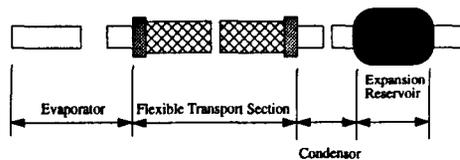


Figure 3.5-7. Flexible Heat Pipe

Critical suggestions	<ol style="list-style-type: none"> (1) Flexible heat pipe can absorb bending, expansion/contraction, not torsion. Pre-torsional stresses during installation may lead failure. (2) By using flexible heat pipes, heat will be transferred locally at where the evaporator/condenser ends locate. It will be more efficient to store heat into few locations by means of phase change heat storage (paraffin), and then use flexible heat pipes to transfer heat to radiators or fins. (3) We should request custom design from manufactures[6] to add a "diode" function to the heat pipes. This is to prevent heat to flow back to the system when $temp_{ambient} > temp_{storage}$.
Sizes available	1' to 4'

3.5.6.3 Radiator and Fins Designs

References	[13],[14],[15]
Techniques	the earth model: Force convection at fins Free convection on plates the lunar model: Radiation to a black body
Critical suggestions	(1). Blowers for force convection must have enough static pressures to keep fins from dusts.

59

(2). The front direction of a radiator should point to the black bodies i.e. dark sky or shadow of Daedalus. Note that the sky is not always dark as well as shadows is not stationary. Some means of rotating the radiator to the right direction is suggested. Otherwise, more than one radiator must be installed so that at least, one radiator points towards the right direction.

Design equations Heat transfer by free convection is influenced by many factors:

$$Q = 0.00414 \frac{A C P^{0.5}}{L^{0.25}} \Delta T^{1.25} \quad 3.5-10$$

A : surface area normal to flow ΔT : ambient to surface temp.

P : altitude to sea level pressure ratio

L, C are characteristic length and shape factor respectively which are based on the physical configurations of the plate surface. These values are suggested in the reference[15]

The temperature of the radiator is a design parameter which must be compatible with the heat input and the physical configuration of the radiator. The temperature can be calculated from:

$$T^A = \frac{QR_{net}}{\sigma} \quad 3.5-11$$

R_{net} is a function of physical parameters which can also be found in the reference[14].

Sizes available No commercial information at this moment. It is expected that radiators and convection plates will be custom design. Fins will with heat pipes.

3.5.6.4 Insulation for Thermally Fluctuated Environment

References	[10]
Techniques	Insulation by material with lower thermal conductivity and low Thermal diffivity
Critical suggestions	Note that thermal conductivity affects the rate of heat transfer.

60

To minimize dT/dt , we need materials with low diffusivity. According to table 3.5-4, foam, usually used in domestic insulation, is not the right material for our problem. Elastomer should be selected instead. However, the weight of elastomer must be taken into consideration. Combination between fiber glasses and elastomers may be the best selection.

Materials	Conductivity W/m.K	Diffivity m^2/s
Engineering alloys	9 to 600	1×10^{-6} to 1×10^{-4}
Engineering ceramics	1.5 to 200	2×10^{-7} to 4×10^{-5}
Porous ceramic	0.2 to 7	1×10^{-7} to 1×10^{-6}
Engineering composite	0.3 to 1.3	1×10^{-7} to 3×10^{-7}
Engineering polymers	0.2 to 0.7	9×10^{-8} to 3×10^{-7}
Elastomer	0.08 to 0.2	3×10^{-8} to 1×10^{-7}
Polymer foams	0.01 to 0.2	1×10^{-7} to 1×10^{-6}

Table 3.5-4 Thermal Conductivity and Diffivity

Design equations We can use the heat equation which has solution in term of travelling waves:

$$\frac{\partial T}{\partial t} = a \nabla^2 T = \frac{\partial^2 T}{\partial x^2} \quad 3.5-12$$

t : time ρ : density
 c : Specific heat a : thermal diffusivity = $K/\rho c$

The solution is in the form of travelling wave:

$$T(x,t) = T^0 e^{i(\omega t - kx)} \quad 3.5-13$$

where $k = k_r + jk_i$. The time of travelling is

$$t = \frac{S^2}{2a} \quad 3.5-14$$

S is the thickness of an insulator. For example, we use 1 cm thick elastomer with $a = 7 \times 10^{-8} m^2/s$. It will delay the time of 12 minutes

61

before the sudden change in temperature reaches maximum inside Daedalus.

Sizes available Any size[9]

3.5.7 Examples of NASA experiences with Heat Pipe Systems

(1) Ames Heat Pipe Experiment(AHPE)[8]

- Temperature control for the on-board processor, electronics packages
- Control gas is nitrogen
- Performance: the system is able to control $20 \pm 5^\circ C$ for more than 6 years.

(2) The Advanced Thermal Control Flight Experiment(ATCFE)[8]

- Demonstrate the long term-term temperature control capability by a thermal diode.
- Control gas is ammonia
- Performance: the system is able to control $90 \pm 2^\circ C$.

3.5.8 Concluding Remarks

Daedalus will be able to perform its intelligent activities as planned only when its electronics are functioning properly. One of the most important factors which affect the performance is temperature. More powerful circuitry means higher temperature inside Daedalus. The steady state temperature rise can be approximately calculated from the power(P):

$$\Delta T = \left(\frac{P}{T.I} \right)^{0.8} \quad 3.5-15$$

The current plan for power usage in electronics parts is about 200 watts (See also chapter 9). This will give the steady state temperature rise as high as $64^\circ C$ which exceeds the working temperature range of electronics. The temperature rise even gets higher when radiated heat from environment is taken into consideration.

In this chapter, we have proposed to use passive techniques using heat pipes, fans, fins and radiators to minimize such a ΔT . These techniques will provide reliability, low weight and functionality to the thermal system.

3.5.9 References

- [1] W.J. Lamon and J. R. Wertz, "Space Mission Analysis and Design" 2nd edition, Microcosm, Inc and Kluwer Academic Publishers, 1992
- [2] F. Horz, "Lava Tubes: Potential Shelters for Habitats", Lunar Construction, Experimental Planetology Branch, SN4, NASA/Johnson Space Center, Houston, TX 77058
- [3] W.S Janna, "Engineering Heat Transfer", PWS Publishers 1986
- [4] D. P. Seraphim, R. Lasky and C. Y. Li, "Principle of Electronics Package", McGraw-Hill 1989

62

- [5] A. B. Bar-Cohen and A.D. Kraus, "Advances in Thermal Modeling", Hemisphere Publishing Corporation, 1988
- [6] Commercial catalogs from Noren Product Inc. 1993
- [7] P. D. Dunn and D. A. Reay, "Heat Pipe", Pergamon Press 1982
- [8] "Heat Pipe Design Handbook-Volume I, II", U.S Department of Commerce, National Technical Information Service.
- [9] McMASTER-CARR, catalog 95 P.O. Box 440 New Brunswick, NJ 08903-0440
- [10] M.F. Ashby, "Material Selection in Mechanical Design", Pergamon Press 1992
- [11] S. Kalpakjian, "Manufacturing Engineering and Technology", Addison Wesley 1989
- [12] S. Kalpakjian, "Manufacturing Processes for Engineering Materials", Addison Wesley 1985
- [13] E. A. Avallone and T. Baumeister III, "Marks' Standard Handbook for Mechanical Engineers", McGraw-Hill 1987
- [14] C. A. Harper, "Handbook of Electronics Packaging", McGraw-Hill, 1988
- [15] J. L. Slon, "Design and Packaging of Electronic Equipment", Van Nostrand Reinhold Company, 1985

4 Navigation Sensors

4.1 Design Rationale

In the current phase the focus of our team has been to identify the possible sensor modalities, which could be used in a lunar or earth based mission and evaluate each one based on their potential usefulness for such a mission. Throughout the document references to the earth and lunar missions are used interchangeably unless specifically noted.

Although the particular design decisions were based on the expertise and experience of the team members a number of cost measures were used to characterize each modality and thus make a more informed decision. The most important and generic ones are presented here:

- **Relative reliability**
The capability of fault tolerance is central to a mechanism which is going to function autonomously in a hostile environment. Therefore single-point failures should be avoided as much as possible
- **Power consumption**
Low power components have a direct effect in prolonging the useful mission time of the robot
- **Computing power**
Computing power is a crucial commodity too, since only onboard computation is to be used to achieve a high degree of autonomy
- **Design criteria**
In order for the robot to carry out it's mission several performance goals must be achieved. This in turn poses constraints in the stride and speed of the mechanism and related constraints on the field of view and the resolution and accuracy of the sensors

Finally in this stage we believe that it is more appropriate to present the criteria which are going to be used rather than describe any alternative solutions which we have already formed.

In the following paragraph the various sensing tasks are identified and analyzed in depth.

4.2 Sensing Tasks

The first step in the design of the perception system is to identify the different tasks that the system will be called upon to fulfill. Later on this information can be used twofold:

7. Identify the most tight and loose constraint for each task and therefore simplify the evaluation of the cost measures.
8. Can be used as a basis of communication between the perception system and the other modules of the overall system, even in the design phase.
9. The tasks which the sensing system must aid are:
 - **Map building**
The robot will need several different levels of terrain maps in order to navigate on the lunar or the earth surface. These will range from local terrain elevation maps to be used in foot placement to higher level mission maps used for long-term planning and navigation. The perception module would probably have to merge several views and range images in order to produce such maps on demand from the other modules. The required resolution and field of view are two of the most important driving factors for this task.
 - **Position Estimation**
The robot will need to know and maintain it's position and orientation on the lunar surface in order to successfully navigate towards points of interest such as lava tubes. Furthermore the robot will have to know explicitly it's position in reference to the it's position before the

last step, in order to perform such tasks as map generation, data fusion, planning and navigation

- **Lava Tube detection**

To perform an efficient lava tube search we need two forms of tube detection. At the first stage the robot is given hints as to where a number of possible lava tube locations are using a long-range detection scheme. At a later stage the robot is able to actually verify the existence of a tube at a possible site.

- **Teleoperation and mechanism survey**

Although the robot is designed to function with the greatest degree of autonomy, a human operator should always be permitted to intervene and affect the decisions of the system in extreme cases. Experience gained from many previous robot designs implies that the operator should be able to have a representation of the environment around the robot in form more easily used by humans than the one the robot will employ for its own purposes.

Finally an important factor in the design is the configuration in which the sensors are mounted on the robot. Since the mounting is strongly related to the sensors and the specific tasks, sensor configuration will be examined independently for each task in the specific section. In the following sections each task is presented in more depth and various different solutions are compared.

4.3 Comparison of Map Building Sensors

For the purpose of designing the perception system a paradigm of three levels of maps is used. Their main difference is the resolution and the area which they cover as well as the required accuracy of the representation. The following maps are used:

1. **Footfall placement**

It is the most detailed and most local map. The resolution of this map is determined by the footfall placement system. In terrain where obstacles are relatively sparse the resolution is taken to be in the order of the size of the mechanism. In more difficult terrains, or when an impossible situation has been detected a more detailed map will be needed, possibly with resolution in the order of the size of foot.

2. **Path planning**

It is more coarse than the previous map and it is customarily used to identify the free-space corridors in which the robot should remain when crossing the terrain

3. **Mission map**

It covers the entire area that the robot will have time to survey. It can be either in the form of a birds-eye view, or in the form of an elevation map. It is used to identify large obstacles, such as mountains and large craters and even possible lava tube sites.

It is evident from the above description that the map building subsystem is composed of three modules:

- **Range sensors**

- **Elevation map generation**

It processes the raw data delivered from the sensors and prepares the elevation map which will be used for later reference

- **Map maintenance**

This component is able to create a local map on demand, either by using the sensors (through the previous module) or by combining previously created maps.

Finally we should note that due to the amount of data that are needed to generate a map this component of the perception system is expected to be the most expensive in terms of computing power.

Therefore a great deal of emphasis will be given in the most time-efficient solution, but not in the expense of the other cost measures like power consumption.

4.3.1 Range Sensors for Map Building

Three paradigms of range sensors were considered: stereo video camera configurations, 2-D laser rangefinders and structured light projectors. For each one a brief description of the advantages and disadvantages is given. We consider that it exceeds the purpose of this document any detailed comparison between different algorithms or products by specific vendors. Our purpose is to evaluate the sensor modality rather than the sensor itself.

4.3.1.1 Stereo camera configurations

The main design decision in a stereo system concern the number of cameras that should be used and the geometry in which they should be mounted. The most well explored cases are two or three cameras mounted in a single line, or 5 cameras mounted in an L-shape or in a cross-like shape. The increased number of cameras is balanced by the reliability and accuracy that such a system offers, in the expense of the computing power that is used to process the stereo pairs and also the decrease of the field of view of the overall sensor (field of view is the intersection of the fields of view of all the cameras).

The advantages of a stereo system are:

- Low power consumption
video cameras are passive sensors, which works by measuring the incident light energy in comparison to the laser rangefinders which have to project their own light
- low cost that leads to redundancy
- camera reliability
space qualified cameras have been in use for a long time
- general in use
a camera of the stereo system can easily double as video input for a human operator

The disadvantages of the stereo system are:

- Computationally intense
although this seems to be the biggest disadvantage of the stereo system, we found that the time-constraint is not as tight as the power constraint and therefore stereo is a promising solution
- Needs light
stereo systems use video cameras which being passive sensors cannot cope with the shadowy areas; as a result the elevation maps will have unexplored areas. A solution could be to carry an additional light source to be activated especially in such cases.
- Difficult calibration
since the performance of the stereo algorithms is very sensitive to the calibration of the cameras, specifically for the lunar mission it is essential to either guarantee that the calibration of the cameras will not be affected by the earth to moon journey or that an automatic way of calibration is included in the system.
- Small field of view
It might be the case that the field of view would need to be enlarged by a pan or pan and tilt mechanism in order for the system to cover efficiently the area around the robot.

Finally specifically for the lunar mission the cameras would have to use dynamic CCD chips, in order to cope with the intense brightness contrasts between the shadows and the lighted areas that the moon offers.

4.3.1.2 Laser Rangefinders

The current technology of the 2-D laser rangefinders involves complicated mechanical scanning mechanisms which reduce the reliability of the sensor. Lately a rangefinders which use optical methods to scan have been made available, either in the form of laser radars or by using optical diffraction grids and electro-optical prisms to direct the laser beam at a specific point in the image.

The advantages of the rangefinder are:

- Speed and computing power
This is their main advantage. They are fast and accurate without consuming computing power
- Works in shadows, night and in caves
Rangefinders are active sensors and they do not rely on the ambient light

Their disadvantages are:

- High cost and large volume which leads to no replication
- Power consumption
Since they are active sensors they have to project their own light. This is their main disadvantages for our mission
- Reliability
Since they are difficult to replicate and they are composed of complex mechanical and optical parts they may be a potential single-point failure.
- Lower resolution (than stereo)

Specifically for the lunar we have to note the lack of space qualified rangefinders.

4.3.1.3 Structured light

The recovery of depth information by projecting a structured light pattern on the scene, like light-stripes, is mature enough to be considered an alternative technology. Their main disadvantages is the power consumption and the limited depth of field of view as well as the fact that the ambient light will complicate their function and possibly create noisy data.

Their advantages are:

- Works in shadows, night in caves
- generic use
as with stereo can be used for teleoperation
- less intensive computation
(compared to the stereo)

Their disadvantages are

- Limited field of view
- Ambient light
- Power consumption

4.3.2 Configuration

In designing the sensing system the mounting of the sensors will be influenced by the desired performance that the robot should achieve and the required accuracy and energy we are permitted to spend in order to achieve it.

For instance a solution in which the robot must move fast would require a broader field of view. In cases when the robot must climb a slope an additional pan and tilt mechanism could be used to compensate for the loss of field of view.

In particular the width of the field of view is influenced primarily by the width of the free-space corridor that the robot needs in order to walk on the terrain and also by the ruggedness of the terrain, since a more difficult terrain would require the exploration of more alternative paths.

On the other hand the depth of the field of view is governed by the length of the stride of the mechanism and the length of path that the robot needs to plan ahead of time, being dictated by the maximum speed of the robot. In addition the depth is governed by the maximum slope that the mechanism is permitted to traverse (in the case where no tilt mechanism is used).

Alternative configurations which we are considering include:

- Sensor mast, which would provide unobstructed view to the surroundings as well as a view of the mechanism, with the additional complications of the initial deployment of the mast and the vibrations during normal operation.
- Pan or Pan and tilt head, which would permit fast and energy efficient coverage of the environment without having to move the body of the robot; also it could easily compensate for the occasional slopes and could be used by the operator to get a idea of the robot surroundings.
- Side mounting, for certain inexpensive sensors, like video cameras, it is easier to mount multiple copies on the periphery of the robot than to employ a special 'head' mechanism.

Specifically for the lunar mission the mounting of the sensors is influenced by the fact that they may need thermal insulation in order to function, and therefore they would have to be mounted inside the body, as opposed to a sensor mast or a pan and tilt head.

4.4 Elevation Map Generation

It was decided that cameras would be the sensors chosen to do map building due to their low cost, low power consumption and durability. In particular the maps will be built with a multi-baseline stereo camera system. An area based shape from stereo method was chosen because these methods determine absolute depth and can be adjusted to produce different resolution images.

The exact configuration of the cameras has yet to be determined. Software exists for configuring multi-baseline stereo systems given parameters like desired resolution, field of view and depth of field. This software can be used to determine an optimum configuration for the stereo system. Camera head positioning mechanism (pan/tilt heads...) need to be investigated as well.

4.4.1 Multi-Baseline Stereo

The generation of a local depth map from the stereo image pairs will be done with the multi-baseline stereo algorithm presented by Okutomi and Kanade¹. This algorithm implements an area based stereo matching method which is performed by computing the sum of square-difference (SSD) between stereo pairs. The results from all of the SSD's are then added to form the sum of SSD (SSSD) to eliminate false matches and increase precision. Features on the lunar surface are difficult to define so an area based algorithm as opposed to a feature based algorithm was used to generate the dense depth map. This algorithm was implemented for the Erebus project using 3 cameras.² The Erebus implementation assumed that the epi-polar lines of the stereo pairs coincided with the scan lines in the images and only checked a fixed number of disparities (pixel shifts along epi-polar lines) in order to speed up the algorithm. The depth map has unnecessarily high depth resolution

1. Masatoshi Okutomi and Takeo Kanade. A Multiple-Baseline Stereo. Technical Report CMU-CS-90-189, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, 1990.

2. Bill Ross. A Practical Stereo Vision System. To appear in proceedings for CVPR '93, NY, NY, June 1993.

for objects close to the sensors and poor depth resolution for objects far from the sensor, so these disparities were chosen so that they made the depth resolution more linear.

SSSD Area Based Stereo Algorithm

1. FOR each row and column in the right image
 - 1.1. select a window around pixel (row,col) of right image
 - 1.2. FOR each disparity value to be checked
 - 1.2.1. FOR every other image

COMPUTE the sum of square differences, SSD(disparity) between the window in the right image and the window centered at (row,col+disparity) in the other image

SET SSSD(disparity)= SSSD(disparity)+SSD(disparity)
 - 1.3. SET disparity(row,col) to the disparity at which MIN(SSSD(disparity)) occurs

Figure 4.4-1 Area Based Stereo Algorithm based on SSSD

Roughly, the speed of the algorithm used on the Erebus project (on a Sun Sparc II) to compute a disparity map for an image with R rows, C columns and D depth bins is,

$$2.5 (RCD) \mu s \tag{4.4-1}$$

so 8 (512x240x27) images could be computed per minute.

A description of a depth map based on rows, columns and disparities in an image is not very intuitive for the person designing a mobile robot. The disparity image (row,col,disparity), with rows and columns being measured from the center of the image, can be converted to a depth map of the local terrain (X,Y,Z) centered on the sensor system with the following equations.

$$X = b \frac{col + \frac{d}{2}}{d} \quad Y = b \frac{row}{d} \quad Z = \frac{bf}{dp} \tag{4.4-2}$$

b = BaselineInMeters
f = FocalLengthInMeters
d = DisparityInPixels
p = PixelSizeInMeters

Physical descriptions of the resolution of the depth images can not be made without setting the parameters and configuration of the camera system. If typical values are used for the camera parameters and configuration then an idea of the physical resolution of the depth image can be found. A typical set up is three cameras arranged in a horizontal line with a baseline of 1.0 m, a focal length of 8.0 mm and a field of view of 50 degrees. The resolution of the depth map for this camera set up at 5.0 meters from the sensor with a minimum depth of 3.0 meters is given in Table 4.4-1.

Image Size (RCD)	Resolution (XYZ)	Time to Compute
256*240*16	6cm*6cm*60cm	2.46 s
256*240*64	4cm*4cm*40cm	9.83 s
112*480*27	3cm*3cm*30cm	16.60 s
512*480*88	1cm*1cm*10cm	54.11 s

Table 4.4-1 Depth map resolution for a typical camera configuration.

The generation of the depth map from stereo is the most computationally intensive part of the perception system. Only 6 depth maps with a resolution of 4cm*4cm*40cm can be made per minute. With the speed requirement of 10m/min one image would be taken every 1.6 meters. This will not generate sufficient coverage of the terrain at an appropriate resolution for navigation purposes, so this algorithm must be speeded up.

Speeding up the generation of the disparity map could be done by limiting the range of disparities to be searched when finding the best disparity for a pixel. A simple technique that limits the range of the disparities to be searched is to put a maximum disparity limit on the disparity map. Because of the inverse relationship between disparity and depth, this translates to a minimum depth on the depth map. The minimum depth could be set to the distance between camera head and the maximum height of the terrain predicted from the previous image. In almost all cases this would insure that the sensor system detects any and all obstacles. The minimum depth could then be converted to a maximum on the disparity search.

Another method for predicting disparity makes the assumption that the terrain imaged by the cameras lies roughly on a plane. The plane equation is assumed to be the same as the plane fit to the previous elevation map. Given this plane, a predicted disparity map can be generated by converting the plane into image coordinates. Each elevation grid point on the plane can be converted to its corresponding (row, col, disparity) point in the predicted disparity map. This prediction map can then be filled in using a median filter or grassfire transform. The stereo algorithm of Figure 4.4-1 can then be executed except that only disparities around the predicted value from the plane assumption are searched.

Another more complicated way to speed up the generation of the disparity map involves predicting the motion of the pixels in the image based on the motion of the robot.¹ When the robot moves the position of a point in the terrain will move with respect to the sensor coordinates according to the transformation $(X', Y', Z') = T(X, Y, Z)$. The movement of this point in the images can be predicted by inverting the equations in Equation 4.4-2.

$$d = \frac{bf}{Z_p} \quad row = \frac{Yf}{Z_p} \quad col = \left(\frac{X'}{b} - \frac{1}{2}\right) \frac{bf}{Z_p} \quad 4.4-3$$

The disparity value given above will not be totally correct, but it will give a disparity for (row,col) which can be searched around until the correct disparity is found. This will limit the number of disparities that have to be searched for each pixel thus speeding up the generation of the depth map. The speed increase in the algorithm will be the number of disparities checked after prediction divided by the total number of disparities that could have been checked.

Disparity Prediction Algorithm

1. CREATE an image called **predicted**
2. FOR each pixel (row,col,disparity) in the previous image disparity map
 - 2.1. COMPUTE (X,Y,Z) in robot coordinates
 - 2.2. COMPUTE $(X', Y', Z') = T(X, Y, Z)$ based on the movement of the robot
 - 2.3. COMPUTE the new predicted (row',col',disparity') of the terrain point (X',Y',Z')

1. Larry Matthies, Takeo Kanade and Richard Szeliski. Kalman Filter-based Algorithms for Estimating Depth from Image Sequences. International Journal of Computer Vision, 3: 209-236, 1989.

- 2.4. STORE disparity' in **predicted** at the new (row',col')
3. INTERPOLATE **predicted** with a median filter or grassfire transform to fill in pixels that have no predicted disparity.
4. COMPUTE disparity map as in Figure 4.4-1 except instead of searching all of the possible disparities for (row,col) only search those around the disparity predicted and stored in **predicted** at (row,col).

Figure 4.4-2 Algorithm for predicting disparity from robot motion.

The speed of the generation of the depth map is also determined by the resolution of the depth map. Limiting the resolution of the map to the minimum required to traverse the terrain will decrease the number of matches made and increase the speed of the algorithm. This minimum resolution will be dictated by the local planner. Fortunately, this algorithm is scalable to a range of resolutions, so maps can be built with the appropriate resolution for the terrain that the robot is navigating.

4.4.2 Converting Depth Maps to Elevation Maps

The disparity map needs to be converted to an elevation map that can be used for local planning and navigation.. Various methods were investigated for generating a dense elevation map from the disparity map.

A grid based elevation map can be generated from the disparity map using the locus method proposed by Kweon and Kanade.¹ This method is advantageous because a dense interpolated elevation map with error estimates is generated for every grid cell in the elevation map. However, once the method was investigated for use with stereo images it was decided that it was too time consuming to be useful. The resolution of the disparity maps is so low that accurate and time consuming methods should not be considered for the generation of the depth map.

The method finally decided is less time consuming and simpler. It involves converting every pixel in the disparity map to robot coordinates by first converting the disparity map to sensor coordinates then converting sensor coordinates to robot coordinates. The uncertainty of the point in the disparity map is converted to an uncertainty in robot coordinates for each point. Each point is then placed in its appropriate grid cell in the elevation map. If more than one point exists for a grid cell the points are merged according to their uncertainties. The final step in the algorithm is to fill in the gaps in the elevation map using a grassfire transform. The algorithm in Figure 1.4-3 contains more details.

1. In So Kweon and Takeo Kanade. High Resolution Terrain Map from Multiple Sensor Data. IEEE Transactions on Pattern Analysis and Machine Intelligence, 14(2): 278-92, February 1992.

Elevation Map Generation Algorithm

1. FOR each point in the disparity map (r,c,d)
 - 1.1. COMPUTE the position of the point in sensor coordinates (X_S, Y_S, Z_S) in using Equation 4.4-2. The error in the depth map is $\sigma_{Z_S} = \frac{1}{B} Z_S^2$.
 - 1.2. COMPUTE the position of the point in robot coordinates using $(X_R, Y_R, Z_R) = T_S^R(X_S, Y_S, Z_S)$.
 - 1.3. COMPUTE the error in the elevation in robot coordinates using $\sigma_{Z_R} = \frac{H^2}{B^2} Z_S^2$ where H is the height of the sensor coordinates above the robot coordinates.
 - 1.4. FIND the grid cell in the elevation map that contains (X_R, Y_R) and
 - 1.5. IF the grid cell is empty store the Z_R in the elevation map and the error on Z_R in the uncertainty map.
 - 1.6. ELSE Update the elevation value and uncertainty with the equations

$$Z_{new} = \frac{\sigma_{old}^2 Z_R + \sigma_R^2 Z_{old}}{\sigma_{old}^2 + \sigma_R^2} \quad \sigma_{new}^2 = \frac{\sigma_{old}^2 \sigma_R^2}{\sigma_{old}^2 + \sigma_R^2} \quad 4.4-4$$

2. GROW each empty grid cell using the grassfire transform until a non empty gridcell is reached. Make the the elevation and uncertainty values of the empty grid cell the value of the non empty grid cell just reached.

Figure 4.4-3 Algorithm for generation of an elevation map from a disparity map.

This method generates an elevation map in the robot coordinate system which is contains information about local placement of obstacles and corridors irrespective of the global position of the robot. For the task of merging maps a method for globally referencing the local elevation maps needs to be determined.

4.4.3 Storing Elevation Maps

Once the elevation map has been created it will be kept around for a while in a map queue. The map queue will be a list of pointers to maps that have been created recently. The purpose of the queue is to provide reference to previous maps for merging.

The maps will have a header which will contain information relevent to the creation of the map including

- A pointer to the position of the elevation map in memory.
- A pointer to the corresponding disparity map in memory
- The pose of the robot in global coordinates.
- The transformation between the current robot pose and the pose of the robot when the previous map was made.
- The time at which the map was made
- The coordinates of the bounding box of the map which will be $(MAX(X_R), MAX(Y_R))$ and $(MIN(X_R), MIN(Y_R))$.

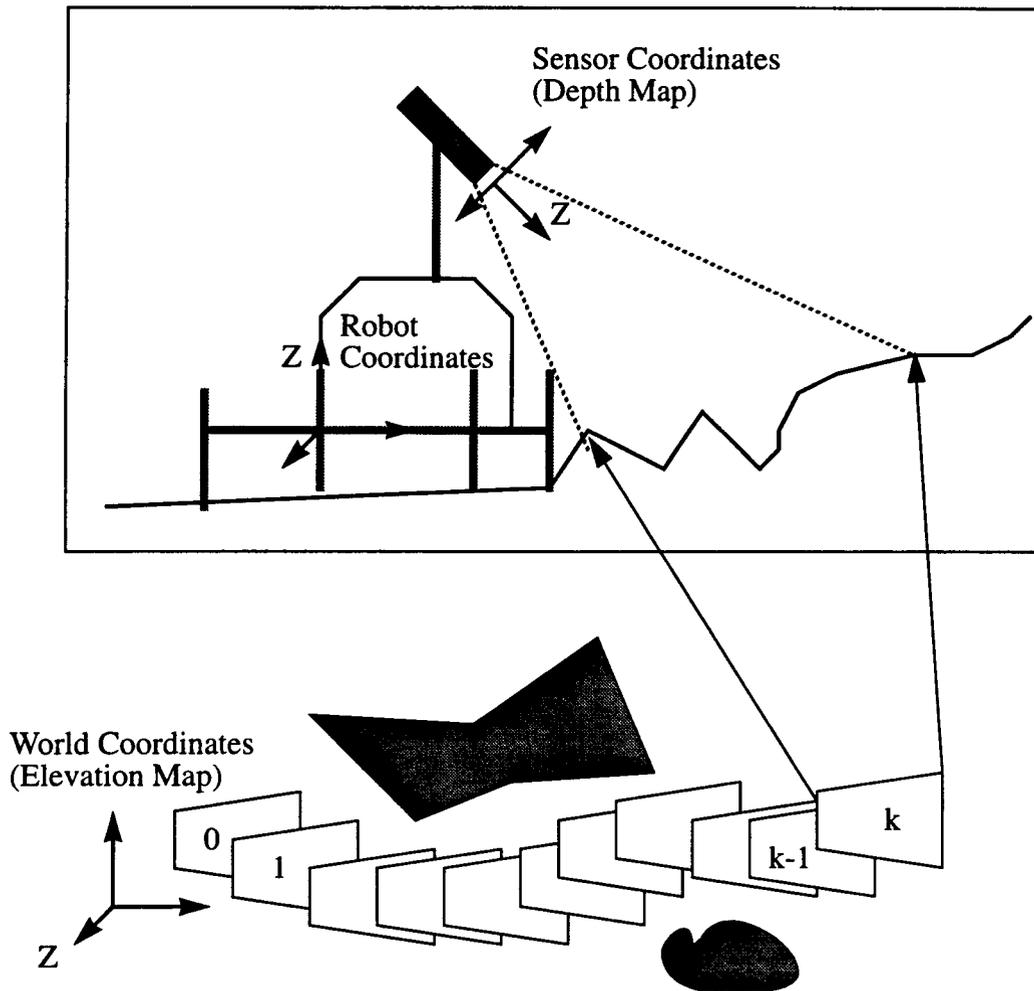


Figure 4.4-4 Reference frames for perception and navigation of the Daedalus robot.

4.4.4 Control of the Elevation Map Generator

The local planner will be the controller for the map building module. When the local planner wants a new map it will tell the map builder a point in the local coordinate system around which it wants a map and the resolution at which it wants the map. This information will be processed and the map builder will make one of three decisions.

- Make a new map at the desired resolution by taking new image(s) of the region. This will involve a transformation of the point and resolution request to camera system parameters (i.e. the pan/tilt/height or selection of cameras of the camera system will have to be calculated from the local planner request).
- Access the desired map from map memory. The region to be investigated may already have been mapped, so a new map does not have to be calculated. The desired map may also be created from a merging of multiple maps in memory.
- Reply that a map around the point of interest could not be generated.

The resolution of the elevation map generated will be the best resolution possible that is less than or equal to the resolution requested by the local planner. In this way the elevation map generator will provide the local planner with as much information as it can. The only way the elevation map

generator will not make a map if the point of interested cannot be imaged and cannot be mapped from merging of previous maps.

Having the local planner tell the map generator the resolution and region of the map separates any planning decisions from the perception module. This modularity will simplify the perception system and keep all the reasoning about the terrain within the local planner.

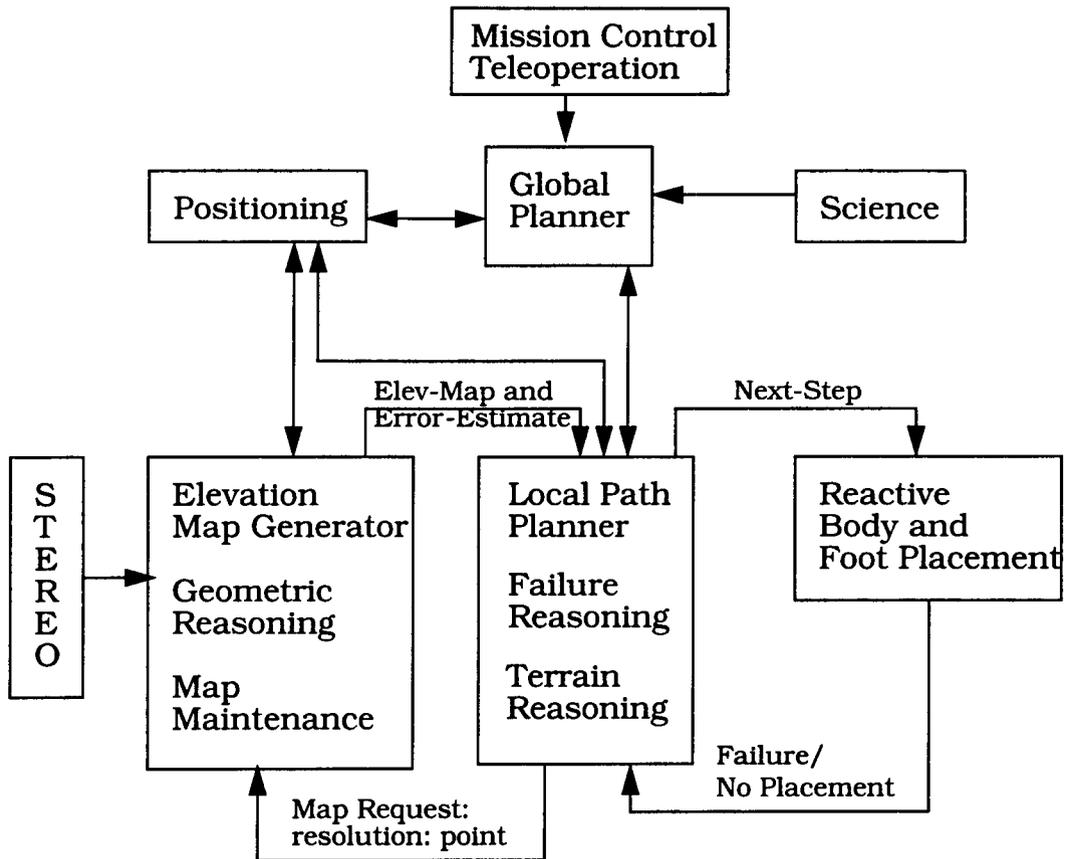


Figure 4.4-5 Block Diagram of Planning and Perception for Navigation

4.4.5 Merging Maps

Occasionally, it will be necessary to combine maps taken in previous positions in order to increase the resolution of an elevation map as requested by the local planner. The local planner request for a map is in the form of an (X,Y) point in robot coordinates at the center of the map and a desired resolution for the map.

Merging Maps Algorithm

1. FOR each map in the queue check if the requested point (X,Y) lies in the bounding box of the map.
 - 1.1. IF the point lies in the box then put the map in the merge queue.
 - 1.2. ELSE do nothing with the map.
2. CALL the map to be made from merged maps **merged**.
3. FOR each map in the merge queue
 - 3.1. COMPUTE the transformation from the current robot position to the position when the map was made by compounding the transformations between maps from the current map to the map being considered. Also compound the uncertainties in the transformations.
 - 3.2. FOR each grid cell in (X_M, Y_M, Z_M) in the map
 - 3.2.1. FIND the grid cell in **merged** that contains (X_M, Y_M)
 - 3.2.2. IF the grid cell is empty store Z_M in the **merged** and the error on Z_M in the uncertainty map for **merged**.
 - 3.3. ELSE Update the elevation value and uncertainty of **merged** with the equations

$$Z_{new} = \frac{\sigma_{merged}^2 Z_M + \sigma_M^2 Z_{merged}}{\sigma_{merged}^2 + \sigma_M^2} \quad \sigma_{new}^2 = \frac{\sigma_{merged}^2 \sigma_M^2}{\sigma_{merged}^2 + \sigma_M^2} \quad 4.4-5$$

Figure 4.4-6 Algorithm for Merging Maps

4.4.6 Calibration

Another issue to deal with is the calibration of the camera system. The journey to the moon will probably jostle the components of the perception system which will bring the system out of calibration. Methods need to be developed that will automatically calibrate the camera system so that accurate depth maps can be generated. One possible avenue is the placement of targets on the robot that the cameras can image in order to calibrate themselves.

4.5 Comparison of Position Estimation Techniques

At first, an estimation of a global position of the robot (i.e. x, y location of the robot in a global map) is discussed. A local position estimation by a dead reckoning system will be discussed later in this chapter.

4.5.1 Global position estimation

In order to determine the best strategy for position estimation of the robot, we have to consider limitations which are posed by the environment, in our case, on the moon surface. If the robot is assumed to be used on the Earth, global positioning systems (GPS) might be the easiest and best positioning method. However, it is not likely that we can use a GPS system on the moon surface which relies on satellites in orbit around the Earth. (More information is needed to decide if GPS systems are really impossible on the moon.)

Several global position estimation methods can be categorized into two groups, vision-based localization and non-vision-based localization.

4.5.1.1 Vision-Based Localization

Vision-based localization is a difficult and challenging problem, particularly in large-scale outdoor domains. Therefore, other types of global localization methods such as the global positioning systems (GPS) should be used when they are available. There, however, are many environments where those special global localization methods are not available, and only vision-based localization method can be used. Those vision-based localization methods can also be used in conjunction with the other type of global localization system, resulting in better accuracy and reliability.

The localization methods in this group include triangulation using landmarks, skyline matching using *a-priori* elevation map and local elevation map matching to the global elevation map.

- **Landmark-based localization**

If several landmarks whose global locations are known, the robot position can be determined simply by using triangulation. The Ambler uses this method to calculate its global position by observing landmarks (i.e. trees) in the global map. Information is needed about the resolution of the global elevation map, how well features can be detected in either camera images or local elevation maps and the accuracy of this method in the case of the Ambler.

- **Skyline matching using *a-priori* global elevation map**

Under certain environments, the positioning by skyline matching have several advantages. Locations of robots can be determined fairly easily if there are clear features found on skylines such as buildings. However, we cannot expect distinct man-made objects on the moon surface. Features which can be possibly used for this method on the moon surface are high mountains, in other words, steep elevation changes. Therefore this method will not work well in fairly flat areas. We still need to know the resolution of the global elevation map and the distance to the skyline on the moon surface (determined by the moon curvature).

- **Local elevation map matching to the global elevation map**

It is possible to match a local elevation map to a part of a global elevation map, using some sort of search technique. This method can give us positions of robots up to the accuracy of map resolutions. However, this method fails if there is no distinct feature in the local elevation map, which will happen often on the moon surface. We still need the resolution of the global elevation map and the resolution and range of the local elevation map.

Signal-based methods and feature-based methods:

Two general approaches to this type of localization is possible. They are signal-based methods and feature-based methods.

In signal-based methods, actual images coming from sensors are correlated to expected images generated from *a-priori* terrain models which are often elevation maps and intensity images. This types of approaches work best when there are good estimates of current location, reducing the combinatorics associated with correlation over viewing position and direction. The critical point of this approach is generation of sufficiently accurate expected images to be correlated to actual images. The signal-based methods has been most successful in military application such as cruise missile guidance where active sensing is used to obtain actual images. This type of images can easily synthesized from *a-priori* elevation map and imaging models.

On the other hand, feature-base methods first extract salient features from the map and the available imagery and then match these features rather than the raw data itself. This approach has its advantages when distinctive landmarks are present and where the available sensing modalities are such that photometrically accurate synthetic views cannot be produced.

Landmark-based localization¹

The landmark-based localization method is explained here in more details.

The method establishes correspondences between visually distinct landmarks and topographically or culturally distinct map features and then infers a viewpoint based on the geometric constraints imposed by these correspondences. Difficulties of this method come from ambiguity in establishing correspondences and combinatorial complexity due to the large number of potentially relevant features that are often present.

To deal with the difficulties, groups in University of Minnesota and University of Utah are studying activities of human expert map users when they localize themselves in unfamiliar outdoor environment by using maps. The groups has built an expert system which can determine global locations, based on the results of the study.

Computational analysis, computer simulations, and experiments done with expert map user all point towards a small set of strategies being critical to the solution of difficult localization problems. These strategies include the followings.

- Concentrate on the view first.
Usually, expert map users try to concentrate on the view first and find distinctive features in it. Then, they make efforts to features in maps corresponding to those features in the view. This makes the search for correspondences more efficient because, in general, much more features are present in maps than in the view. Those maps usually cover an area much larger than can be seen. In other words, features in the view are far more likely to be relevant than features on the map, since most map features will not be visible from any single viewpoint. Information about the immediate surroundings can significantly constrain possible viewpoints. As a result, the search for feature correspondences should start from the view.
- Landmark features should be organized into configurations.
Correspondence is aided when individual features are assembled into configurations of multiple features. These configurations should have both viewpoint invariant properties that help in searching over the map and viewpoint dependent properties that constrain possible viewpoints.
- Information about terrain at the viewpoint is important.
If it is possible to determine that an agent is at or near a particular terrain feature type, then the viewpoint is constrained to be at one of the corresponding types on the map. Determination of local feature type is often easier than evaluation of more distant features.
- Multiple hypotheses need to be generated and examined.
In general, terrain features are highly ambiguous, so that it is difficult to identify landmarks with certainty. Any single viewpoint hypothesis based on a small number of features has a high probability of being incorrect. Therefore, we need to develop a number of different hypotheses for the viewpoint before verification takes place.

1. References

- W. B. Thompson, H. L. Pick Jr., B. H. Bennett, M. R. Heinrichs, S. L. Savitt and K. Smith, "Map-Based Localization: The "Drop-Off" Problem," In *Proc. DARPA Image Understanding Workshop*, pp. 706 - 719, 1992.
- W. B. Thompson and H. L. Pick Jr., "Vision-Based Navigation," In *Proc. DARPA Image Understanding Workshop*, pp. 149 - 152, 1992.
- S. L. Savitt, T. C. Henderson and T. L. Colvin, "Feature Extraction for Localization," In *Proc. DARPA Image Understanding Workshop*, pp. 327 - 331, 1992.
- M. H. Heinrichs and etc., "Strategies for Localization," In *Proc. DARPA Image Understanding Workshop*, pp. 321 - 325, 1992.
- K. T. Sutherland, "Sensitivity of Feature Configuration in Viewpoint Determination," In *Proc. DARPA Image Understanding Workshop*, pp. 315 - 319, 1992.

- Hypotheses should be compared using a disconfirmation strategy. Validation of a hypothesis involves a comparison of the actual view with expectations generated from the map based on the presumed viewpoint. In performing this comparison, it is the most important to note expectations that are not met. If one clear mismatch is found, then the associated hypothesis should be eliminated.

Univ. of Minnesota and Univ. of Utah group studied how human expert map users determined their locations in unfamiliar outdoor environments. As a result, it is found that all expert users who successfully solved the problem employed four strategies out of six described above. Those strategies include:

1. Concentrate on the view first.
2. Landmark features should be organized into configurations.
3. Multiple hypotheses need to be generated and examined.
4. Hypotheses should be compared using a disconfirmation strategy.

This indicates that a landmark-based global localization method can employ these strategies to increase efficiency of search for correspondences without losing reliability.

Other components which are necessary for landmark-based localization methods include sensitivity analysis in viewpoint determination and extraction of map and image features.

Sensitivity in Viewpoint Determination

After correct correspondences between visible landmarks and map features have been obtained, the viewpoint must be determined. The viewpoint, in principle, can be calculated by a simple triangulation. However, effective localization also requires an understanding of the errors that can occur. The knowledge about the error involved in determination of the viewpoint can help us to choose the best landmarks, so that the viewpoint can be determined with the least error. Distance estimates to landmarks will not be available often. It is also common that measurements of absolute bearing to a single landmarks or the visual angle between landmarks have some amount of uncertainty associated with them. This uncertainty propagates through the localization computation and translates into uncertainty about the actual viewpoint.

By observing several examples, it can be seen that the nature of the error is very substantially with the particular configuration of landmarks used.

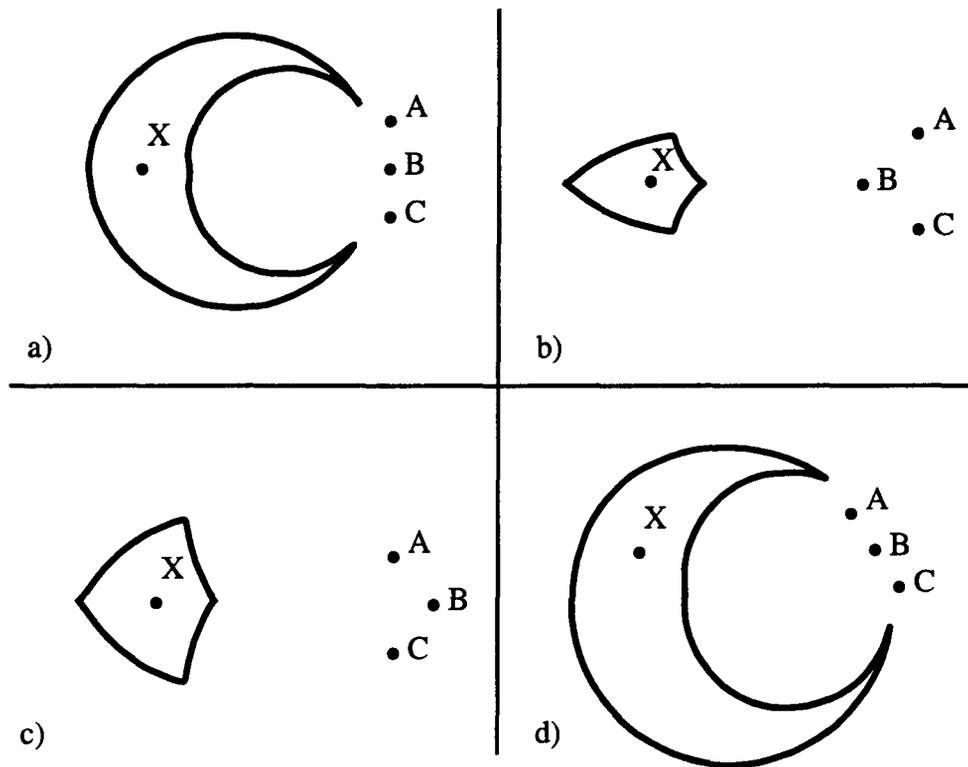


Figure 4.5-1 Sensitivity in Viewpoint Determination

Figure 4.5-1 shows four examples. In the figure, X marks the actual observation point. A, B and C are the locations of three landmarks. The dark line shows the region of uncertainty associated with an error in the estimation of the visual angle between landmarks. In all four cases, the observation is at the same distance from the landmarks, and the landmarks are at the same distance from each other.

Figure 4.5-1a and Figure 4.5-1d shows the situation when landmarks are in a linear configuration, in other words, all landmarks are located on a straight line. This type of configuration results in larger uncertainty of the viewpoint localization. If the viewpoint is at a position offset to the side with respect to landmarks (Figure 4.5-1d), the area of uncertainty is skewed away from the true viewpoint. In general, non-linear configuration (i.e. all landmarks are not on the same straight line.) reduces the area of uncertainty.

Overall, it is important to choose the configurations that lead to the least uncertainty, when there is a choice of landmarks available which are used for determining the viewpoint.

Extraction of Map and Image Features

Making correspondences between image features from the viewpoint and map features requires two tasks: a process of extracting map features which are corresponding to image features and a process of detecting distinct image features.

Important landmarks in outdoor environments include peaks, ridgelines, saddles, and valleys, and these features have precise definitions in terms of differential operators applied to underlying surfaces. Those operators are known to work fairly well in the case of point features such as peaks and saddle points in digital elevation models. However, extracting line features in a map with its elevation map given by using those local operators is usually very difficult, so that we need alternative methods to extract those landmarks reliably.

The group of University of Utah and University of Minnesota proposed a new method to extract line features such as ridgelines and valleys. The method uses a hydrologic simulation to determine

the flow that would result if the local topography is used with a uniform density of fluid. High flow rates in the simulation are associated with valleys which are reasonably distinct in a given elevation map.

4.5.1.2 Non-Vision-Based Localization

Possible localization methods in this group include observation of the robot from the Earth using laser, the use of startrackers and beacons.

- **Observation of the robot from the Earth using laser**
The distance from the Earth to the robot on the moon surface can be measured by using laser. This measurement was used for measuring the distance between the moon and the Earth up to the accuracy of 12 cm. The only required equipment on the robot is a small mirror (about 30 x 30 cm). However, there are several questions about this method. We do not know if we can measure directions to the robot from the Earth which give us (x, y) locations on the moon. It is also not clear how long the measurement takes and whether the facilities for the measurement are really available to us or not. We still need to obtain a NASA documents on this measurement.
- **Startracker**
Startrackers used on the Earth might be available on the moon surface. It seems that orientations can be determined by the device, but we doubt that it can calculate global positions. We need to find a document on startrackers.
- **Beacons**
Global positions can be determined by using beacons located in known positions in a global map. If the beacons' position in the global map are unknown, we can determine only relative positions to those beacons. The problem is how to place the beacons after a landing on the moon surface. (It will probably be impossible to place beacons during the landing.) We need to get documents on commercially available beacons (if they exist).

4.5.2 Local position estimation

Dead reckoning is a widely used method that identifies the position and location of a robot by integrating its movement history. A weighted least-squares approach to dead reckoning for legged mechanisms which was developed for the Ambler is available for estimating the APEX robot. This method has already been tested on the Ambler. It is reported that the systematic error increases by two percent of the traveled distance, for the specific configuration of the Ambler. The source of the systematic error is believed to be incomplete kinematic model of the Ambler mechanism, which can be improved for the APEX robot. On the other hand, slips between the robot's feet and the ground and backlashes of the robot joints can increase the error for the APEX robot because the robot's weight is much less than the Ambler's, resulting in a less rigid structure. (The Ambler's prismatic joints can be controlled within millimeters, and the rotary joint backlash is less than a hundredth of a radian.)

The accumulation of errors created by dead reckoning can be partially halted by matching objects from elevation map to elevation map. With Kalman filtering the uncertainties of local transformation matrices can be improved without referring to the global map. The transformation matrix from the global reference frame to the robot frame can also be calculated in less uncertainty if exact locations of landmarks are known in the global maps. We still need to know the resolution of the global elevation map and the resolution and range of the local elevation map.

4.6 Teleoperation and mechanism survey

At times it will be necessary for a human operator to intervene in the progress of the robot. Intervention might be done for:

- Global feature detection
- Searching for scientifically interesting features
- Inspecting failure of the robot

This intervention will involve the sensing system, so ways to incorporate it into the perception system must be investigated.

5 Planning and Navigation

5.1 Overview

The communications flow between modules follows a simple philosophy: higher levels do not control lower levels, they just advise. This leads to robust behavior, since lower level modules generally have a better idea of what's going on than more abstract levels. The more control the modules closest to the real world have, the more robust the system.

Our global route planner uses the global position estimation to monitor the robot's position within a global map. It gives advice in the form of a direction to the local planner that will avoid large known bad areas in the a priori map. The local planner combines this advice with knowledge of the surrounding terrain gotten from the perception system to give more precise directional advice to the reactive planner to help it avoid large obstacles in the immediate vicinity. The reactive planner then moves the vehicle as best it can. This is not demand driven. The lower levels continue acting on advice until new advice comes down.

As in subsumptive architectures, this philosophy gets around the common problem of having to precisely act out a plan laid down from on high. It makes the system driven by the environment rather than driven by presuppositions.

5.2 Global Route Planning

The purpose of the global navigator is to get the robot from a start position to a goal by the best route possible, given the information at hand. Global route planning is often neglected in robot missions since most robot missions are of a "local" nature: The robot is placed at the site and it performs a task such as toxic waste removal or exploration. Daedalus will have to traverse across country, traversing large distances while surviving the hazards of open terrain. What does a global navigator gain us? If the robot has well defined destinations, a global navigator can get us to that destination, avoiding dead ends and dangerous areas that a purely "local" robot might fall into. Without a global map, a robot is doomed to wander towards its goal, probably relying on humans to give directions and protect it from the pitfalls.

We need two things to do global navigation: an a priori map to build a plan, and a way of pinpointing our position during the route execution. If we have these two things, there are a variety of ways to plan and execute the global route which we will evaluate.

5.2.1 Maps

What kind of map should we use? Global route planners in the past (such as that used by the Hughes cross country system) have used digital terrain elevation data maps (DTED maps). DTED maps are straightforward to create, just send a plane over the area to be mapped, get a stereo pair, and use it to produce a DTED map with a resolution of 5 to 10m per pixel. Unfortunately, these maps do not contain enough information. They give high density terrain elevation information, but what about more semantic information such as information about roads and stretches of water? A stretch of water, such as a lake or river looks like a good place to send a robot. Is it? At the resolution of a DTED map, a road looks just as good (or bad) as a stretch of meadow. Do we want to send the robot down the road or through the meadow? Does this decision depend on the road type, which is also not in an elevation map?

Another type of map that the military uses for planning is the Interim Terrain Data (ITD) map format. Instead of representing the terrain as pixels, it represents the terrain using polygons and segments. ITD format uses several different maps (or themes) to represent the same stretch of terrain. There is no direct elevation data, but there is a "surface configuration" theme, which gives polygons of homogenous slope. There are also themes for transportation networks (roads, bridges,

and railroads), for surface materials (soil, clay, etc.), for vegetation (what type and how high), and for drainage (rivers, streams, and lakes).

A major problem with the ITD map is availability. The creation of an ITD map is not automated. Parts of it, such as the surface configuration theme, could be, but aren't. What happens is that a DTED map is collected, and then each theme is annotated, by hand, on top of the DTED map, using whatever other information or expertise is available about road type, vegetation type, drainage, etc.

If we cannot get full ITD data for our target site, we can do the following: Get the DTED map, and then make an ITD-like map that might not contain all of the information in the full ITD-format, but would still contain enough information to plan with. The same thing could be done if the robot ever went to the moon: get the elevation map, and then annotate regions that the experts think would be bad for the robot to go into.

5.2.2 Global Positioning

We will not address the "how" of global positioning in this section, but rather we will address the "why." Why do we need global positioning? On Earth, the most common way for robots to find out where they are is to use the Global Position Satellites system. GPS has been ruled out, because it won't work on the moon. Unfortunately, no other proven method has ever been implemented on a robot, and development and implementation of such an autonomous method within the time constraints of the program might be impossible. If it is impossible: can we do without global positioning?

There have been suggestions that maybe the robot doesn't really have to know where it is to do global navigation. After all, people manage to navigate on a global scale without having to refer to satellites, why can't the robot? Tod Levitt's work on Qualitative Navigation, or QualNav has addressed this problem. QualNav is an attempt to replicate the human ability to navigate without ever really knowing where you are.

The intuitive idea behind QualNav is that instead of navigating based on your knowledge of your position, you navigate based on your knowledge of landmarks. It assumes that you can see landmarks and estimate distance and angle to them. Instead of monitoring your position, you monitor your topological relationship with the landmarks that you can see. Thus a global route is specified by a sequence of topological relationships, such as, stay left of mountain 1, go between mountains 3 and 4, etc.

The most plausible situation for QualNav is if you have a robot with very bad dead reckoning exploring a world with many easily recognizable features. The topological relationships between features can be added to the "map," and the world can be re-traversed later. Off course, there will be distortions in this map due to the bad dead reckoning, but the power of the topological representation will keep the effect of these distortions down to a minimum. Now, is that the case with our robot? No. We will have an a priori metric map. What QualNav would have us do would be to take that metric map, extract landmarks, create a topological network with them, and then throw away the original metric map.

The major problem with using QualNav for our situation: It doesn't make the problem any easier. It assumes that you can detect range and angles to features. This is the hard problem that we would be trying to avoid by using QualNav! If you can do this (and no one has ever demonstrated a reliable system that does it), you can use this information to position yourself within you metric map, thus avoiding the stated problems of QualNav! Actually using this information is the easy part of the problem. If you can get the landmark information there is absolutely no reason to use QualNav.

If we don't have a global positioning system there is no need to work on global route planning, since the robot will have to work entirely locally, i.e., instead of following a trajectory planned to avoid bad places, the robot will just head off in a direction chosen by human operators and keep going in that direction until a human intervenes. The main risk with this approach is that the robot will get into a situation it can't get out of, or will explore many dead ends before getting to its goal. If these risks and liabilities are acceptable, then we can do without global navigation entirely.

5.2.3 Global Navigation

We have all of these knowledge sources (the themes of the ITD maps), how do we find a route through them from a position to a goal or goal region?

5.2.3.1 Finding a route

Whenever the problem of route planning comes up, the first answer to look at is the A* search. The A* algorithm promises to give the optimal path, if you give it the right heuristic functions. The only knock on the A* algorithm is the amount of real time and space it takes, but since the route will not be planned that often, this is not too much of an issue.

So, we have information in terms of polygons and segments from the ITD maps. We could do a search through the raw ITD maps, but the problem of finding out the possible "next" moves could be intractable. We suggest using the method used by Chuck Thorpe: Rasterize the maps into pixels, and then assign to each pixel a cost of traversing it. Thus you could have a large cost of traversing high slope pixels, and a very low cost for traversing road pixels. The possible next moves are trivial to generate: the surrounding pixels.

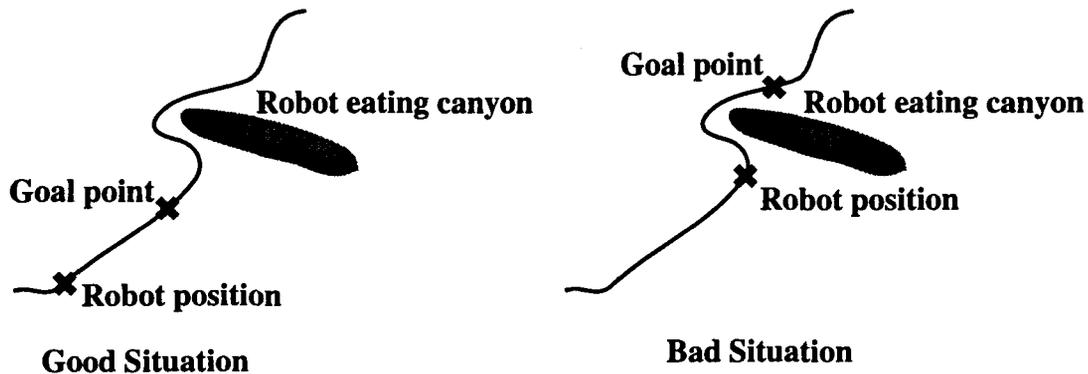
This rasterization of the theme maps provides another advantage: It isolates the route planner from the type of map used. Say we use ITD maps in development, but in the field have to use our own map format. We could rasterize them both into the common format that will be used by the route planner.

Now the obvious thing to do is to use the A* algorithm is to use it to provide the one true route and then attempt to follow it. This has several problems. First of all, the A* algorithm tends to produce routes that hug obstacles. After all, the A* algorithm produces the optimal path, i.e., the path which is shortest, which, of course, is the path that takes the vehicle closest to the obstacles. Secondly, it is naive to assume that the route we give to the local planner will be followed perfectly. The local planner has to be able to take precedence over directions from the global. The global map could be wrong, or the position in the map could be wrong.

5.2.3.2 Following a route

If we generate one route for the robot to follow we will have to balance staying on the path to safely get to the destination with allowing the robot to deviate from the path for obstacles or other anomalies. One simple method is to just pick a point in along the path which the robot will

aim for. If the point is far enough away, then the robot will be free to deviate laterally, while still following the route. Unfortunately this simple method leads to an obvious problem:



The method works in the good situation, but in the bad situation the local obstacle avoidance would have to work very hard to get over the global route planner telling it to head into the robot eating canyon.

A better idea is to use a potential field approach with two forces. One force would impel the robot parallel to the path. This force decreases as the robot moves away from the path. Another force would suck the robot back on the path, and this force would increase as the robot moves away from the path. The combination of these two forces will bring the robot back to the path when it deviates and bring it along the path when it is on it. We can even imagine incorporating forces from “obstacles” to reduce the obstacle hugging tendency of the A* search.

Unfortunately, all of these methods concentrate on planning a single path, and then following that path. What happens if the robot deviates so much from the path that getting back to the path is just not a good option? At what point do you replan? What is the criteria for replanning?

5.2.3.3 Who needs a route?

We can use the approach suggested by the Hughes Cross Country System: why bother with a single path? Instead of calculating the “one true path,” start from the goal, and work your way out to find the best way to the goal from any point in the map. This “gradient” function can be found with the same A* search mechanism that produces the single path. A simple outline of the algorithm would be to start searching from the goal towards the starting position. Instead of following the pointers back from the start to the goal to find the one path, retain all of the nodes in the A* search. Each node can be used to generate a “best direction” at a grid cell. The nature of the A* search is that the nodes will be the grid cells most likely to be visited by the robot.

Here is a more complete specification of the algorithm:

```
open list is a grid with ordering kept by a 2-3 tree, initially containing
  only the goal states.
closed list is a grid.
```

```
while (node = Best from open list) {
  delete node from open list
  insert node on closed list
  if finished, return
  get successors of node
  for each successor {
```

```

    if successor on open list and successor has better score than
      the node on the open list
      replace it.
    else if successor on closed list and has better score {
      replace it
      propagate change through closed list
    } else add successor to open list
  }
}

```

This is essentially a standard A* algorithm. Each node corresponds to a map cell. The formula for the A* score of a node is

$$\text{Score}(\text{node}) = G(\text{node}) + H(\text{node}) \quad 5.2-1$$

Where $G(\text{node})$, i.e., the cost to get from the goal position to the node's grid cell, is defined by

$$G(\text{node}) = \sum_{\text{all nodes in path}} 1 + \text{terrain score} \quad 5.2-2$$

Essentially this sums up the costs of all the map cells from the goal to the node, plus the number of cells in that path. The heuristic function, H , is simply defined as the Euclidean distance from the node to the start cell.

Two things make this different from a standard A* search. First of all, the algorithm does not simply return the optimal path from the goal to the start. The purpose of the algorithm is to evaluate all the map cells that the robot is likely to reach when traveling from the start to the goal. Thus, the algorithm returns the entire contents of its closed list. This difference in purpose also leads to the second difference: The search does not stop when the start cell is reached. The search continues for a while, thus exploring more regions where the robot might end up. The extreme case is to allow the search to explore the whole map.

Once we have the closed list, how do we compute gradients. We could just take a look at the "next best" pointer to determine the next best direction, but that would lead to a very coarse gradient map, with only eight possible directions. A better method, also gotten from the Hughs system, is as follows:

```

For each node in closed list {
  Do N times {
    Find best next node.
    If direction of node changes by more than 90 degrees, exit.
  }
  Use position of current best node to establish the gradient
  direction at the current grid cell.
}

```

This algorithm results in a finer gradient field. By increasing N you can get more precise gradients, but as N increases, the accuracy of the gradients reduces. We address this problem to some extent by stopping the loop when a drastic gradient change is detected.

This algorithm could be fairly expensive in terms of memory. Let us take a straw-man example. Assume a 12km x 12km map, with a cell size of 20m by 20m. This comes out to 360,000 cells. Cost maps could grow to consume megabytes, but consuming megabytes is not a big deal on a modern computer with up to 32M of RAM.

If memory becomes a limiting factor, then a hybrid solution can be used: Use A* to find the optimal path from start to goal, and then subdivide this path into large cells. Each cell will then

have the gradient calculated for all points, using the point where the path enters the cell as the start and the point where the path exits the cell as the goal. This will bring back some of the original problems of following the "one true path," but if the cells are large enough, the problems will not be too severe.

How long will this algorithm take? Let us put five minutes as the limit for reasonable computation time. Since the global route planning only has to be done once, spending five minutes of one day out of a multi-day mission does not seem like much. Assume that we run the algorithm to completely cover a 360,000 node grid. This is about 830 μ s for each node. For a Sparc 2 with around 30 MIPS, this is a lot of time.

If you do not allow the planning to evaluate the whole map, then you will have to deal with the situation where the robot is forced into an area of the map which was not covered in the planning. One option at this point is to start up the search again and let it progress until the cell that the robot is in is covered. This has the advantage of being correct, but it could be very expensive in terms of time and the memory needed to keep the whole search structure around. Other possibilities include doing a simplified A* search from the unexplored cell toward the goal until you hit an evaluated region.

Not doing complete coverage of the map will save some initial planning time, but could cause problems in case the robot wanders outside of the covered area. If you are going to do complete coverage of the map, why not use a breadth first search instead of an A*. The only change to the algorithm would be that the open list would become a LIFO queue instead of a 2-3 tree, and the algorithm would go from being an $O(N \log N)$ problem to being an $O(N)$ problem, and increase in speed of at least an order of magnitude for $N=360,000$.

Even if you could do complete coverage in a reasonable amount of time, it would seem that the space requirements would be prohibitive. Not really. Incomplete coverage means that much of the information used by the A* search, such as cost estimates, best next cells, etc., must be kept around for each map cell. If you do complete coverage, all that has to be kept is the gradient direction, so it is not clear that incomplete coverage definitely uses less memory than complete coverage. If you do complete coverage, you never have to worry about the computational cost of augmenting the coverage if the robot starts seriously wandering.

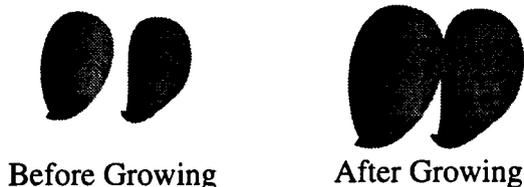
5.2.3.4 Improving the map

Both the A* coverage and the BFS coverage will cause the robot to hug obstacles. This is because both approaches try and find the "optimal" path. The optimal path is essentially the shortest one, i.e., the one which hugs the obstacles the most. Continually skirting the edges of obstacles in the name of optimality is risky.

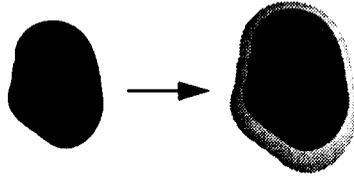
We suggest preprocessing the map to grow the obstacles. Here is a naive algorithm:

```
do in parallel for each map cell c
  for each neighbor n of c
    if cost n < cost c then let cost n = cost c
```

When we say, "do in parallel," we of course mean simulate the effect of doing the loop in parallel. This algorithm has the unfortunate effect of eliminating possible routes through the map, as the figure shows.



A smarter algorithm doesn't simply grow the obstacle cells, but extends their influence. What we want is an algorithm that produces something like this,



When the map is changed in this fashion, the planning algorithm will have sensible results. When faced with a choice between traveling between two bad regions and going around, the robot will probably be told to go around, but if the only way through is between two bad regions, then the robot will go through there. An algorithm to do this expanding will look something like this:

```
do in parallel for each map cell c
  for each neighbor n of c
     $\text{cost } n = \text{cost } n + \text{weight} * \text{cost } c$ 
```

The variable **weight** should be chosen to be small enough that new "insurmountable" obstacles will not appear in the map.

5.2.3.5 Updating the plan

A problem with the algorithm is this: What if the map is wrong? What if the local planner consistently sends the robot in a direction greatly different than the direction indicated by the global gradient? This would imply that the gradient map is not giving good advice, and should be changed. This implies two problems, deciding what the changes to the map should be, and then changing the gradient map accordingly.

One way to decide what changes should be made would be to keep a history of differences between gradient direction and robot direction. When the error threshold is reached, i.e., the global planner decides that the robot has been swimming against the gradient stream for too long, the planner will look back over the history and place virtual obstacle pixels in the map corresponding to obstacle pixels that would cause the deviation.

Once the changes have been made, we could just completely replan, but that would be horribly expensive. There is the possibility of doing an incremental update. The scheme is this: start with the pixels that you change. Look at the surrounding gradient pixels. For each one whose gradient has a component that points toward the changed pixel, negate that component. After you are done with those pixels, look at the pixels adjacent to the new changed pixels.

This heuristic addition will not give the same result as recomputing the whole gradient, but it will be a lot faster, and intuitively it makes sense, i.e., it results in pointers away from the new obstacles, and does not change things that do not point at the new obstacles.

5.3 Local route planning

The local planner spans the gap between the higher level modules of global map-based planning and mission control and the lower level modules of perception and reactive control. Given an elevation map created by the perception modules, and a desired heading from global planning, an efficient obstacle avoiding path is generated. Lower level modules execute this path reactively and are free to deviate from the path as long as they notify the local planner of their final position. Higher level modules advise lower level ones in a loosely coupled manner and the local planner continues moving the robot in the last specified direction until informed otherwise. If lower level modules

are unable to execute the desired path within the general guidelines, an error is generated and the local planner presents alternate paths. If desired, active perception is used to gather additional information. Repeated failures are reported to higher level modules, and optionally brought to the human users' attention. This section focuses on a viable approach for generating good paths through relatively sparse terrain. In denser obstacle fields, a more deliberative strategy is desired, and this scenario is also explored.

5.3.1 Specifications

This module communicates with the following other modules:

- Global path planner
- Elevation map builder (perception)
- Positioning
- Reactive control

The input to this module consists of a map, represented as grid cells containing elevation information. This will include at least

- Expected z value in the given grid
- Uncertainty information about z value (e.g. variance).

In addition, the local planner is given a heading, either in the form of a direction, or a distant waypoint. This is provided by the global planner. From this information, the module generates an obstacle free path which connects the current robot position to the goal position. The module considers the following when evaluating potential paths:

- Distance traveled
- Leg motions (low cost)
- Body rotations (high cost)
- Body height changes (high cost)

In general, flat paths consisting of long straight segments are preferred over shorter paths which require the robot to crawl over numerous obstacles.

In the case of a sparse obstacle field, the module will output a target cell in the elevation map, and the reactive control module will attempt to guide the robot to that cell. When a more deliberative approach is required, either a sequence of cells, or even explicit motions may be generated. Information about minimum body height is given in any case.

5.3.2 General Philosophy

The general planning philosophy for the planners is that higher level modules should advise rather than dictate actions for the lower level modules. Thus the waypoint or direction given by the global planner will be seen as a general goal rather than a specific destination. If the elevation map indicates that the waypoint is unreachable, a different suitable point will be chosen as long as it lies within some bounds of the original destination. This flexibility is also encouraged in the lower level modules which are free to deviate slightly from the given path in order to perform their tasks more efficiently. The idea is to help the lower level modules by giving them access to knowledge which is not directly accessible to them. However the raw data (e.g. elevation map) is always made available in case the lower level module needs to directly access the information during operation.

5.3.3 Resolution

Since the elevation map will be generated using stereo, it is possible to obtain a variety of resolutions, depending on the time available. However the optimal resolution for each stage of planning is not immediately obvious. The local planning module will require at elevation maps with grid sizes which are on the order of the robot's body size. Grids which are larger than this would rule

out a number of viable paths, whereas smaller grids could increase the time taken by the planner to unacceptable levels.

Ideally the robot should get a new elevation map every footfall. Since the desired speed is 10 meters/minute, and each step is about 30 cm in length, we estimate that the robot will have about 4.5 seconds for each step. We assume that the robot can plan one footfall ahead of perception, so that an elevation map may be generated while the robot is moving. Assuming that an image can be digitized when the robot has just started a step, the following actions must take place before the next step can be started:

- Stereo must generate depth map
- Depth map must be converted into elevation map
- Local planner must find a suitable path

Since the processing will take place on a standard workstation, one sees immediately that the resolution of the elevation map must be rather coarse. In essence, local planning's job is to avoid the larger obstacles, while leaving the task of avoiding smaller ones to the reactive control modules.

The situation is drastically changed when the robot enters a dense obstacle field. At this time grid cells on the order of the footprint, or the triangle frame may be desired. However since the robot is a frame walker, and is thus incapable of planning individual footfalls, it may be true that the additional resolution may only confirm that there is no good foot placement strategy available to us in the given situation.

5.3.4 Input

As discussed above, a coarse elevation map will be needed at regular intervals. An asynchronous coupling is recommended for robustness, however specific elevation maps could be requested by the local planner in specific situations (especially during error recovery). The request will certainly include the following information:

- Desired resolution.
- Position of the elevation map, relative to either the robot or the previous elevation map.

In addition it is recommended that the elevation maps generated by the perception modules record at least the following data:

- Time-stamp
- Position data (from positioning module)
- Unique identifier

Map merging should be decoupled from the local planning module.

Possible templates for the input functions are provided below. Note that these are only guidelines and implementors are encouraged to adapt them for needs of efficiency or hardware change:

- `get_elev_map(coord_struct, resolution_type, &map)`
- `coord_struct` is in the correct reference frame
- `resolution_type` may be an enum for the valid types.

THIS SECTION IS INCOMPLETE — AWAITING FEEDBACK FROM OTHER GROUPS

The direction information from the global planner is received periodically by a polling function. If there is no new direction available, the last given direction is used. Thus the robot continues moving in that direction unless impassable obstacles are encountered.

5.3.4.1 Coordinate Systems

It is important to ensure that the reference frames used by various modules are consistent. A coordinate frame with the following salient features was determined to be suitable for Daedalus:

- The frame is relative to the elevation maps rather than the robot.
- It does not necessarily change with every elevation map received.
- When global positioning is performed, the reference frame may be adjusted to be the current robot reference frame.
- All elevation maps are reported in the current reference frame. This facilitates perception requests and map merging.

5.3.5 Output

After much discussion, it has been determined that the reactive control modules only require a single target point in the elevation map. By exploiting real-time short range sensing, an intelligent control scheme will be able to execute a suitable path for the robot frame from the current position to the goal position.

The desired output in the case of a dense obstacle field is less clear. More investigation into this area is required before any predictions can be made.

5.3.6 Algorithms

The local planner will employ some form of rasterized A* search algorithm. A suitable cost function will be created, which includes the criteria for both a safe and an efficient path. Although the local planner will plan for several footfalls at one time, only the first target point will be reported to the reactive control module. As long as the reactive planner is able to perform adequately no replanning will have to take place. However, whenever the reactive control module places the feet far from the desired location, replanning of the path may be needed.

5.3.7 Error Recovery

The loose coupling between modules should result in a robust system. However the issue of error detection and recovery is more vital in this case since a failure may not be immediately apparent to higher level modules. Each module is responsible for updating its superior, who will evaluate the state of the system based on its knowledge. If it determines that the robot has strayed from its true path, it must replan if possible and try to compensate for the drift. If this is not possible, an error condition for the higher level module is generated. Thus it is possible that an error in reactive control could propagate all the way to global planning. However the more likely case is that numerous errors in lower level modules will conspire to create a situation worthy of higher level interest.

It is instructive to observe the behavior of this module in a specific scenario (**GLOBAL PLANNING: we need to talk!**). Assume that the global module's plan has been lost or was initially incomplete. The local planner is still able to use the coarser or possibly incomplete data. Assume that an unexpected obstacle was encountered and no local solution discovered. This obstacle would be reported to the global planner, and incorporated into the global map. The gradient fields would be recomputed and a new direction would be sent to the local planner. In the extreme case, we can operate with a completely blank map, assuming that we have an idea of the final and current positions by merely heading towards the goal. As obstacles are detected, they are added to the map. Thus local planning in effect performs a real time A* like search with the heuristic of Euclidean distance. It is even possible for the global planner to completely change its map in the middle of a mission with no adverse effects on the lower modules. This is because the local planner has no real notion of state — neither elevation maps nor previous directions are kept. It is thus possible for a pathological local planner to drive Daedalus in an infinite circular loop. The primary

function of the local planner is to detect and avoid visible obstacles, but the current design can be easily expanded to incorporate other relevant tasks.

5.3.8 Dense Obstacle Fields

A dense obstacle field can be defined as one where the time taken to deliberately plan a viable path through the field is smaller than the time it would take the reactive planner to execute such a path. A deliberative planner requires the following:

- An elevation map with sufficient resolution to place feet.
- Sufficient computing power and time to solve the problem.
- Terrain must have enough flat areas for feet.

The terrain condition is worth a closer look: since *Daedalus* is a frame walker, its feet are moved as a unit (unlike the *Ambler*). This means that the terrain must have safe footfalls arranged in the right configuration. This assumption holds on sparse terrain, but as more obstacles are added, this constraint becomes increasingly difficult to satisfy.

Deliberative control requires the planner to be able to see the terrain to a resolution on the order of the foot, rather than on the order of the robot. Unfortunately, rough calculations with the stereo algorithm indicate that this will not be possible given the planned hardware configuration. From this, it looks unlikely that the local planner will be able to move *Daedalus* through a dense obstacle field using deliberative planning.

5.3.9 Simulation

Simulation allows designers to investigate the behavior of the proposed system under a variety of (albeit simplified) conditions. It is recommended that the local planning software be extensively tested in this manner to ensure that the algorithms are able to perform under the desired speed/memory constraints. Initial simulation should consist of artificially generated elevation maps, possibly incorporating synthetic errors. Further testing should be performed in conjunction with the perception module to ensure that the local planner can handle real data equally well. The interface between reactive control and local planning also demands extensive testing since the number of errors flagged by the reactive control has a great impact on performance. However this type of simulation requires a more detailed world model and it may be better to test the system in the field. In contrast deliberative foot fall planning modules are very amenable to simulation and the behavior of the robot under conditions of varying terrain can be seen during simulation.

5.3.10 Conclusions

Since this module is neither firmly connected to the hardware of the robot nor the reality of the sensors, its specifications and goals are rather unclear at times. In the final implementation, it is likely that the local planner will have to glue together the various other modules into a working system. Towards that end, it is important to maintain a flexible architecture, which can easily be extended as the requirements and outputs of the neighboring modules are updated. Generally speaking, the local planning module has the following requirements:

- Elevation maps with coarse grid cells (on the order of the robot) at a fairly high rate. To plan for multiple footsteps, a range of 10 meters seems to be sufficient. Since the map will be updated frequently, it is not as important for the long range data to be accurate.
- Reactive control should be robust enough to handle small problems on its own. Since local planning is already cycling as fast as possible to plan each step, it is unlikely that there will be idle time in which to process complaints and replan.
- Global planning should pick paths which are sparse on obstacles, even at the expense of traveling slightly longer distances.

Deliberative planning has been shown to be of limited usefulness in this scenario because of resolution and configuration constraints. However a deliberative planner is still desirable as a backup for the reactive planner and in teleoperation scenarios. Simulation should not be ignored as an effective means of evaluating various algorithms quantitatively.

5.4 Positioning on the lunar surface

Since the APEX robot will need to locate features of scientific interest on the Lunar surface in order to complete its mission, the robot will need to be able to locate itself on the Moon. A variety of methods for accomplishing this have been proposed, from human operators on the Earth locating the robot and sending it its location, to matching local elevation maps with a priori maps of the Moon's surface without human intervention.

5.4.1 Star Tracking

The method proposed in this section draws heavily upon navigation systems used by sailors on the Earth for centuries, namely navigation by the stars. First I will outline a system that will allow positioning of the robot anywhere on the Lunar surface, then I will analyze the errors inherent in this positioning system.

The system makes use the following four coordinate systems:

- Global coordinate system G: The coordinate system is set up somewhere in the Solar System. Only the orientation of this coordinate system matters, so it doesn't matter where the origin is located.
- Moon coordinate system M: Similar to many coordinate systems used on the Earth. A good system might have the origin at the center of the Moon, the z-axis passing through the Moon's north pole, and the x-axis pointing toward the Earth. This coordinate system should be fixed to the Moon, and rotate with it.
- Robot coordinate system R: This is a coordinate system fixed to the robot, probably with an origin fixed somewhere easily identified on the robot body, with z-axis pointing straight up and x-axis in the direction of motion.
- Camera coordinate system C: The standard camera coordinate system, with the origin at the focal point of the lens and the z-axis pointing in the camera direction.

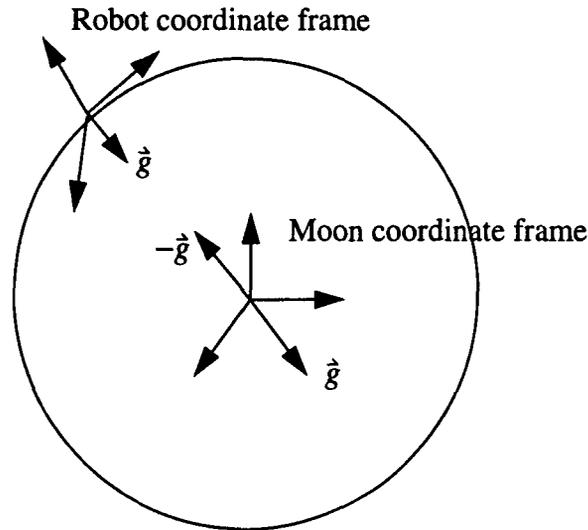
At first, we consider only the orientation of these coordinate systems. Given a camera on the robot that can see the stars, it is possible to construct the rotation matrix R_G^C , from the global coordinate system to the camera coordinate system. Apparently, JPL has a commercial star-tracking package that can determine the orientation of a camera with respect to a global frame in this manner.

Given the exact time, we can construct the matrix R_M^G , the rotation matrix from the global coordinate system into the Moon coordinate system, from ephemeris data.

If the camera is calibrated with respect to the vehicle, we also have the matrix R_C^R , the rotation from the camera coordinate system to the robot coordinate system. This calibration will certainly have to be performed in order to use the cameras anyway.

Given the above rotation matrices, it is possible to construct the composite rotation matrix R_R^M from the moon coordinate system to the robot coordinate system.

Since it is planned for the robot to level itself, the robot will have to have inclinometers to measure the gravity vector \vec{g} , in the robot coordinate system R. If we transform this vector into Moon coordinates using the rotation vector R_R^M , then take the inverse of the resulting vector, we have a vector that points from the center of the moon to the robot, in Moon coordinates. This vector can then be easily converted into latitude and longitude.



The errors in this scheme come from two sources:

- Errors in the measurement of one of the rotation matrices.
- Errors due to the gravity vector, both measurement errors and systematic errors due to the fact that the gravity vector may not point exactly at the center of the moon.

For errors of the first type, the deviation can be expressed as an difference in each of the rotation angles. These errors map directly into degrees of error in the latitude and longitude of the vehicle. One degree of arc on the Moon is roughly 30 km. It is expected that the star tracker and the ephemeris data will have very small errors associated with them, leaving the camera calibration as the largest source of systematic errors in position.

There will be random errors in the measurement of the gravity vector, but these errors should be zero mean, and thus can be integrated out through multiple measurements. An issue of more concern is whether or not the gravity vector really points toward the center of the Moon from all points on the surface, and if not, how large the deviation can be. For comparison, a large gravitational anomaly in the United States produces a deviation of 20 arc-minutes in the gravity vector (that is, the gravity vector is off by 20 arc-minutes from pointing toward the center of the Earth). Deviations of similar size on the moon would generate a systematic error of 167m with this positioning system.

5.5 Positioning on the Earth's surface

Since the global position estimation method described in the previous section will provide similar position information to that of (non-differential) GPS on the Earth's surface, it seems only natural to use GPS for positioning of the terrestrial robot. This is especially true since the star tracking system would not work during the terrestrial daytime, and would suffer atmospheric disturbances if it were to be used at nighttime on the Earth's surface. Since the terrestrial robot will be running during the day, GPS seems a logical choice.

6 Computing and Electrical Hardware

(The Daedalus requires extensive computing resources due to the data processing done on-board. Although detailed computing requirements have not been estimated, based on previous system, the Daedalus may well require 100 MIPS and 25 MFLOPS of computing capability. Two options are currently being considered for Daedalus: the first is based on the Harris developed, MIPS R3000 series processor with space qualified packaging and the second is a ruggedized commercial computer, such as a Sun Microsystems Sparc 2.

The Harris solution offers several benefits: Since it is designed to meet space qualification procedures, the computer is rugged and can survive in a radiation environment. Second, the system has built in redundancy and fault tolerance. Finally, the system is fabricated using advanced multi-die modules, which permit placing an entire system in several square centimeters with minimal power requirements. Harris has also developed multi-chip memory modules with densities exceeding tens of megabytes per cubic centimeter. The major drawback to this solution is its cost, both to purchase and to port existing operating systems and software.

The commercial processor solution is orders of magnitude less expensive than the Harris system and it has the benefit of being well understood by the software developers. However, the computer is physically quite large and consumes a fair amount of power. In addition, commercial processors are not as environmentally rugged as those designed for space application. The proper selection of the processor for this mission will depend heavily on available funding, although the commercial processor may have a great impact on the rest of the system.)

6.1 Control System

The hardware control system provides for basic functions such as actuator control for walking and sensor polling while communication with computing in charge of more complex functions such as perception and planning.

6.1.1 Sensors

There will be several sets of sensors located throughout Daedalus. On the body, there will be sensors that will provide data such as the robots current inclination, internal and external temperature and humidity, and internal power levels. The leg sensors might return data such as whether the foot has made contact with a foreign object, and the current length of a stride.

While most of these sensors will constantly provide information Daedalus, some might be used to trigger the robot that an unexpected event has occurred. For example, the foot contact sensor might trigger indicating that it has made contact with a foreign object that was not originally identified by perception.

6.1.2 Communications/Telemetry

Communicating between Daedalus and the outside world will be by wired tether, wireless ethernet and a wireless low-speed modem.

The wired tether will be used to during the development and debugging stage of the robot. The ethernet connection will be used as the primary means of communication during experiments. The modem running over a SLIP will be used for demonstration purposes of its ability to function remotely.

6.1.3 Computing

The main task for the control computer systems is to control the actuators that gives Daedalus motion. Control tasks will be performed by two Dynaterm CPU boards running on a VME bus. One is dedicated for motion control and the other for asynchronous tasks. These particular board were

7 Telemetry

7.1 Purpose of Telemetry and Tele-operation

The Daedalus robot is designed to be an autonomous frame walker for use in long term missions. One of the eventual goals of the robot is to attempt a lunar mission of extended duration. In addition to survival, it is hoped that Daedalus would report back useful information which it has gathered during its mission. In general, communication with the robot while it is performing its everyday tasks may be needed.

The basic needs of the telemetry system is to provide a clear and error free link between the robot and whatever ground control station is monitoring its progress and discoveries. The link should allow significant information to be exchanged both on the return link from the robot to ground control, and on the forward link from ground control to the robot.

To ensure that the commands are received correctly over the link, the transmission must include significant overhead to account for error detection or correction. On commands being sent on the forward link, it is important that they are received with complete correctness. As a result, a byte of data would likely multiply into two or three bytes of encoded data. When bulk data, such as images or depthmaps are being transmitted, an infrequent error will not be as undesirable. There are several methods of encoding currently in use by NASA, which are often extensions of simple error correction/detection schemes such as Hamming codes or Cyclic-Redundancy Codes.

The information being transmitted over this links are of the high level type. Low level commands, such as commands to motors are bypassed, and the system instead would transmit a command to the mechanism controller. As a result, the telemetry link is not a means of operating the robot, but rather a means of communicating with the modules already on board the robot.

However, along with telemetry, the robot can be tele-operated using the same link. This implies that one or more of the modules are being given supplemental commands or are being completely replaced by an off-board system. In this method, any information generated by the robot which would be sent to the replaced module is instead sent along the telemetry link to the ground control, where it is processed by a simulator mimicking the robots current status, and the output of the simulator is sent back to the robot where it is used by the other modules. Clearly on the lunar mission, this type of procedure will be quite slow due to the time delay and would only be attempted for demonstration purposes or in instances of extremely high necessity.

While the robot is in the development phase, the use of telemetry and tele-operation should play a much greater role than in the final missions. When coupled with the simulator as described in a later section, sample missions and problems should be able to be tested in complete safety to the mechanism.

7.2 Fitting Telemetry into the Architecture

Using TCX, a point to point architecture, and an Ethernet local area network as the basis for the architecture on Daedalus, the telemetry fits in quite easily. The ground control needs to send a packet of information to the robot and have it distributed correctly within the system. Using the wireless ethernet connection or using one of the other link strategies discussed later, the result of the telemetry is the sending of an ethernet packet from the ground control network over a transmission to the on-board ethernet.

Since the architecture is point to point, once an initial socket is established between the destination module and the telemetry connection, the packet will be placed onto the received queue of the destination module. In addition, this module can be instructed to regard only the packets coming from the telemetry socket during a tele-operation run. If the transmission system employed is

not wireless ethernet, the unit which receives the signal can easily dump the received information onto the ethernet network, since most of these units function as an ethernet node.

However, since the normal ethernet packet contains a large amount of redundant overhead bits to allow for addressing on a complex system, one alternative is to use Compressed Ethernet. This concept, which involves reducing the header information given a known and static environment configuration, is being developed by Martin Marietta in work on the UGV project.

An ideal situation would be to connect the relay station to the internet. This would allow for remote operation without the use of satellite intervention and allow for a twenty-four hour mission control setup at CMU. The most likely way for this method to work, would be to access a dial up terminal line and use a standard 9.6 or 19.2 Kbaud modem.

Using the cellular link would make this easy, otherwise the relay station would need to be located near a phone line. Possible help from universities or other research institutions near the chosen site would allow the phone call to be local rather than long distance. However, even using a 24hour AT&T dial-one service, the cost would be roughly a hundred and fifty dollars a day.

7.3 Mission Issues

The design of the telemetry system is very dependent upon the scenario in which the robot operates. Clearly, the needs on an earth mission are considerably different than those required on a lunar mission. On an earth mission, the robot can be monitored from a ground station very near the robot, while on a lunar mission all monitoring will come from satellite links to the robot. In addition, while on the earth, the rules and regulations regarding broadcast frequencies and transmission wattage must be observed.

It is true that the mission affects the telemetry issue in quite a major way, however, regardless of the mission, certain requirements will be constant. One major example is the need to link Daedalus to some form of relay station. This link needs to be untethered to allow the robot the freedom to explore relatively long distances and rugged terrain. The link should also not be extremely dependent upon line of sight transmission, since while performing exploration, the robot could easily traverse into a ravine or over a ridge from the relay station.

The reasons a relay station is being used is quite basic. On the earth mission, the need to actually use satellites to communicate to the control station is minimal, since the ground control station can be located near the mission site, and the priority of a request for satellite event times would be extremely low. Hence, real-time operation and monitoring of Daedalus would be essentially impossible. In addition, if Daedalus was responsible for communicating with a satellite directly, the issues of keeping the link open become very important.

A highly directional beam is usually required to make contact with a satellite and the addition of a servo motor to keep this alignment would increase the complexity and the power required to run Daedalus. With an off board relay station, a short range omnidirectional transmission can be used on board the robot minimizing the complexity of the mechanism, and increasing the probabilities that the robot maintains contact with the monitoring station at all times.

While not required in the earth mission, an extremely useful exercise is to provide a method to link the relay station to a mission control at an extremely far distance from the mission location. The most likely method of transmission for this link involves bouncing signals from satellites in geosynchronous orbit. Using this technique several times during the mission would demonstrate the ability of the systems and modules to be monitored or tele-operated through a time delay similar to ones which would be experienced on the lunar mission.

Another constant of the telemetry not affected by the mission itself, is the bandwidth needs and descriptions of the data stream and control stream. The bandwidth must be high enough to allow useful monitoring of the discoveries made by the robot. The frequency and wavelength used for transmission need to be unaffected by the weather or atmospheric conditions on the earth, as well as solar affects on the lunar surface. The data stream and control stream must provide access to everything which may need to be monitored or changed during the mission.

7.4 Daedalus to Relay Station

There are several different methods of linking the robot to the relay station, and again they are quite mission dependent. The basic methods are a short range transmission using a wireless ethernet link, using a commercial cellular phone link, a short range line of sight transmission over the Ku band, or a medium range transmission using a radio or CATV, UHF, or VHF television channel.

All of these methods have their advantages and all have some disadvantages. Rather than deciding upon one of these methods and using it throughout the development process and mission, it may be more fruitful to use each method when the time is appropriate. The next four subsections describe these three alternatives and give the times when using these technologies would be appropriate, and when they should be forgotten.

7.4.1 Wireless Ethernet Link

Wireless Ethernet is useful for short range communication (up to six miles). However, toward the limit of this distance, the link must have unobstructed line of sight operation. While Daedalus explores rugged terrain, this may not always be possible. Wireless Ethernet was used on the Ambler project with success and should certainly be used during the development and debugging of the robot.

Currently, FRC has a pair of ARLAN 620 WE Bridges. These run about \$4000 each and use spread spectrum transmission (SST) to achieve reliable transmission. They do not require any sort of FCC licensing to operate in the US or Canada. They attach using standard coaxial cable to the ethernet LAN on board the robot and at the monitoring station.

SST was invented for ultra-secure military use to avoid interference, and therefore is reliable in areas with high disturbances as well as providing good data-loss ratios. SST involves spreading the transmission signal over a large frequency band with very low power per unit bandwidth. At the receiver, the signal is compressed back into the narrow, high intensity band normally used in radio transmission.

size	10"x10"x2"
weight	3 lbs.
power	24V DC with an 18 Watt draw
safe temperature	0 to 40 C
data-rate	1000Kbps (observed during ambler about 640Kbps)
frequency	~910MHz
antenna	8" dipole

Table 7.4-1 Technical Specifications of the ARLAN 620

During the use on Ambler, it was noticed that when the Perceptron (a metal encased laser scanner) was situated in between the sending and receiving antennae, the effectiveness of the network was lowered by a noticeable amount. This suggests that in the open terrain, this technology might not be that advisable since total communication loss may occur with an occluded sight line.

7.4.2 Commercial Cellular System

This possibility is clearly only available on the Earth mission, due to current sparse coverage of the lunar surface as a result of low consumer demand. SpectrumCellular markets a product which allows error-free data communication between two sites. Since most of these components are marketed for automobiles, and since the only power available to automobiles are a 12V DC power supply, this technology is very much in line with mobile robot systems.

size	7" x 3.5" x 1.2"
weight	1.25 lbs.
power	12V DC with a 6 Watt draw
data-rate	modem dependent (300 to 9600? available)
CPU	Z-80 with 4kRAM and 16kROM
antenna	8" dipole

Table 7.4-2 Technical Specifications of SpectrumCellular Bridge

In actuality, the feasibility of a phone connection is potentially questionable, since the robot will be conducting a mission in a relatively obscure and sparsely populated region. Also, the cost associated with a cellular contract, which could be quite high, may be negotiable using some sort of advertising or publicity on the mission. The relatively low bandwidth of these modems would also hamper the sending of large size data packets such as images or depthmaps. However, the size and weight and power usage of the device make it very convenient.

7.4.3 Ku-Band Line of Sight Link

Since the area to be explored is relatively small, and assuming that a support vehicle will be relatively close to the Daedalus (typically less than 5 km) at all times, at this close range, an omnidirectional antenna, coupled with a low power transmitter, could be appropriate. The main drawback of this technology is that a directional receiver is required. As a result, the servo motor and controller remarked upon earlier would need to be added to the robot. If this investment is made, then the use of this link becomes feasible, otherwise it is not going to work at all.

This would be a useful technology for the earth mission, unless it is decided not to have a chase team which stays in close contact with Daedalus at all times. During a two week run of over two hundred hours, Daedalus could easily cover a distance of forty to fifty kilometers. On the lunar mission, clearly this technology is inadequate, since it commits the robot to stay within a short distance of a deployed relay station. The alternative is to increase the size of the receiving dish and use the controller and servo to bypass any form of relay station. However, a larger dish could easily get in the way of the sensor mast and the solar umbrellas needed on the lunar mission.

The following is a simple link estimate that shows the viability of the concept. This estimate assumes using the Ku band with an omnidirectional transmitting antenna, 1 W of transmitted power, a 0.2 m receiving antenna with 40 pointing accuracy and a data throughput rate of 10 Mbps using a convolution coding scheme. The data rate of 10 Mbps was chosen because this is the Ether-

net bandwidth, which would alleviate the need to envelop additional hardware/software support. Since a viable margin is typically less than 12 dB, this link is quite robust. Commercial Ku band products with these capabilities exist and can be used for this mission.

Parameter	Value
Effective isotropic radiated power (EIRP)	-2 dB
Transmission path losses (10 km)	-134 dB
Atmospheric and misc. losses	-5 dB
Received isotropic power (RIP)	23 dB
E_b/N_0	16 dB
Coding gain	5 dB
Implementation loss	-2 dB
Margin for BER $1E^{-8}$	19 dB

Table 7.4-3 Estimate for downlink using Ku Band Link

7.4.4 Radio/TV Link

Perhaps the most intriguing method of communication would be the use of a radio-frequency (RF) modem which broadcasts in a range and power which requires FCC licensing. This technology gives non-line of sight communication in distances up to thirty miles (over forty five kilometers). This would be more ideal for a lunar mission, and would be useful on the earth mission should a chase team not be following the robot.

MultiPoint Networks sells a device which is described below. This is one of many devices which are available of this type. While this particular device is perhaps the most expensive in terms of power and size, its range and data-rate are the highest found. The normal 9600 baud 10-15 mile RF modem operates at a lower wattage in a smaller size. The costs of the device and the FCC licensing are unknown at this time. The data rates of the various devices range from 4800 baud up to 64Kbps.

size	17.5" x 15" x 3.5"
weight	unknown
power	115/230V AC with a 60 Watt draw
safe temperature	0 to 50 C
data-rate	64Kbps
frequency	~930MHz

Table 7.4-4 Technical Specifications of the Multipoint RAN-64

transmit power	5 Watts
time delay round trip	~4 ms

Table 7.4-4 Technical Specifications of the Multipoint RAN-64

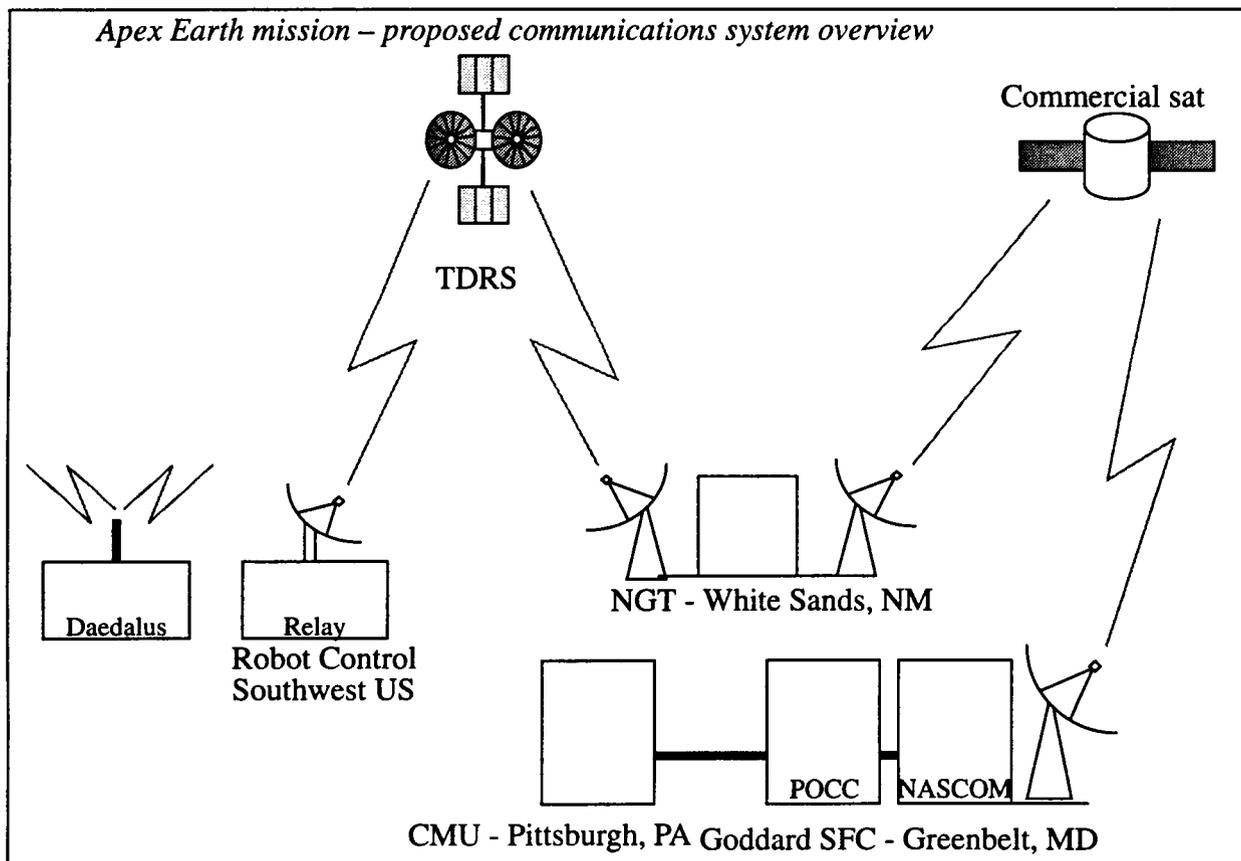
7.5 Relay Station to Mission Control

The communication from the relay station to a Mission control center is not required all the time for the earth mission. However, demonstrating that the robot can be controlled and monitored using a low bandwidth and high time delay link is beneficial to promoting the space readiness of Daedalus.

On the recent Erebus mission, a telemetry scheme was established in connection with NASA. The outline of this method is shown in a figure below. The system provided a transparent TCP/IP link between the relay and the Payload Operation Control Center (POCC) located at Goddard Space Flight Center in Maryland. The forward data rate (to the robot) was 9.6Kbits/sec and the return data rate (to control) was 1.544Mb/sec. The differences in these amounts are due to the channels allocated the project on the Timing Delay Relay Satellite (TDRS). The return rates are generally high to allow for the shipping of video images, etc. on the data stream.

The link had a 2-4 second time delay due to the link passing over the commercial satellite. The video compression techniques used allowed shipment of 5-6 frames/second during low net activity. Audio signal was communicated as well using an additional Inmarsat terminal.

Figure 7.5-1 Telemetry Proposal for Apex Mission (used on Erebus)



7.6 Streams for Data and Control

After determining the link, what is being sent becomes the next issue of importance. The control stream should almost always be of low bandwidth unless some reprogramming or database changes are being made. The data stream will almost always be high bandwidth and very full. This is due to the mission objective being one of exploration. Since the robot won't have much storage space on board, it is important that information be continuously dumped back to a control station.

7.6.1 Data Streams

Each of these streams requires a different bandwidth. Most of the very high bandwidth information is probably easily packaged in less than 100 bytes of data. However, there is other information which should be relayed back to control as well which has a much greater packet size. An example of these are depth maps, stereo images, science payload device results, etc.

When combined with the simulator or with some tele-operation, it should be possible to run the robot using the telemetry information and accomplish a virtual mission. In the event of module failure or other catastrophe, it should also be possible to jump in and replace or tele-operate the dying module or routine.

Below are listed the most likely candidates to be sent back along with their rough size and their estimated bandwidth frequency, the values are summarized into a table at the end of the streams section.

7.6.1.1 Map Position

This few bytes of indices into a global map should take virtually no bandwidth, but will allow for tracking of the robot and for determining when the robot has made a major error. Also, should the robot find an interesting site, having the ability to broadcast its position allows a debriefing team to note this location on an off-line map for future reference. Since this position won't change much, there is no need for this to be a high bandwidth operation and probably one position update per minute or one per five minutes would suffice.

7.6.1.2 Thermal Index / Power Supply

Again, a few bytes of information, and probably not more frequent than once every minute or every five minutes. These allow for determining when the robot will need to have a replacement of batteries, and to notice if something is going wrong with the cooling system. On the lunar mission, this would also be useful to determine if the robot is likely to survive the lunar night and could serve as an indicator to turn on some sort of low power heating device. Since this packet will never be full, any other frequent information should be able to eventually piggyback onto this space without a significant increase in overhead.

7.6.1.3 Science Instrumentation

Several options are available here, as described in the Scientific Instrumentation section. However, aside from high resolution images, most of these results take no bandwidth and function only infrequently. Some continual running scientific instruments will be on board most likely, however, their bandwidth and power consumption will be very small.

7.6.1.4 Logging Stream

The two alternatives here are continuous low bandwidth transmission, or infrequent bursts of high bandwidth dumps of the logging buffer. When the robot is close to a goal, or seems to be acting oddly or is in a dangerous situation, this stream of status logs and other module information should be provided. More than likely the best way to handle this is to dump information in large bursts rather than continually sending smaller packets. As a result, the frequency depends on the size of the buffer and the amount of logging information, but hopefully once per hour should suffice.

7.6.1.5 Images or depthmaps

Using the low bandwidth radio or wireless ethernet links, the number of images which could be transmitted to the relay station preclude any possibility of real time images, so there is no reason to try to send back more than one image per five minutes or so. Since this will eat up much of the bandwidth during that time, and since the most important time to have images is when the robot is being tele-operated over rough terrain, it is clear that tele-operation will always be quite slow.

7.6.1.6 Body positions and mechanical information

A relatively high frequency report, but probably requiring around only about thirty bytes of information is the positions of the encoders on the body and legs, etc. This allows the observation of the reactive systems and should also give an idea of how the gait could be controlled to reduce power consumption, etc. This information should be most useful during debugging or development, and possibly may not be needed during an actual mission at all.

Description	Frequency	Amount of Data
global map position	once every five to ten minutes	five bytes maximum
thermal and power supply	once every minute	five bytes maximum
scientific instrumentation	variable, usually quite low	variable, one kilobyte?
logging stream	once every hour	on order of kilobytes
images and depthmaps	once every ten minutes	on order of kilobytes
mechanical information	frequently during debugging	fifty bytes maximum

Table 7.6-1 Data Stream Frequencies and Data Sizes

7.6.2 Command Stream

Most of these will be very small in size and should be basically instantaneous commands to the robot. Any commands given from the monitor should certainly override those decisions by the robot itself, since it is assumed that the human monitor knows better than the robot and is acting in its best interests.

One of the mission objectives is to provide an autonomous robot for long time periods. As a result, the needs of the command stream are very minimal. During tele-operation however, the needs of the command stream blossom as expected. The need to activate science and the desire to move sensors are the most basic tele-operation features, with module control being perhaps the highest level.

7.6.2.1 Sensor Control

The commands to tilt cameras or to switch to a higher force trigger, etc. should not take up much bandwidth. However, these commands will likely be relatively high priority whenever they are issued. Issuing these sorts of commands while the robot is exploring could result in hindering of progress. Therefore, it might be necessary to have the robot cease movement when the cameras or other instrumentation are being changed by an outside force.

7.6.2.2 Module Control

A relatively high bandwidth operation, running a module via tele-operation may have its uses. What each module does still needs to be specified, however and the bandwidth needs will likely hinge on how often these types of things need to be controlled. Some modules will not be tele-oper-

ational due to the time constraints. Operating under a time delay generated by the lunar mission will render the reactive foot placement completely not tele-operational as one example. There are other modules which would be hampered by the time delays as well.

7.6.2.3 Science Payload Control

The commands to initiate the scientific instrument procedures, or the commands to request reporting of gathered data, like all the other basic command stream operations should require minimal bandwidth. None of the scientific instrumentation needs any real outside information other than commands to begin or to report, so even if the robot is tested in the earth mission without a true lunar payload, evaluation of the command link should not have to take that issue into account.

7.6.2.4 Updates Position, Map, Reprogramming

Any forms of updates to the robot in terms of database or software will likely be very time consuming. However, accuracy during these moments is crucial, so the robot will probably just sit and wait until all the data is transmitted anyway and so all the bandwidth should be usable.

Description	Frequency	Amount of Data
sensor control	only when needed	minimal, few bytes
module control	during tele-operation	minimal, few bytes
science payload control	when desired, possibly automated as well	minimal, few bytes
reprogramming	hopefully never	variable, probably large

Table 7.6-2 Command Stream Frequencies and Data Sizes

9 Power Requirements

Although it is commonly an after-thought in the design process, proper power management is crucial to the success of any robotic endeavor. This is especially true for a robot such as Daedulus that is designed to ultimately function for long periods of time without human interaction. Unfortunately there is no turn-key system available that one can purchase which would handle all power management. Power management is more involved than just combining a couple of batteries together: It requires the careful combination of power storage and generation with thorough consideration for the requirements of all systems and possible mission objectives to successfully complete its mission.

In the initial stages, the earth bound version of Daedulus will require some physical human interaction but in its full incarnation Daedulus will be autonomous. Careful planning of power usage and rationing will therefore be as important as the system implementation itself, if not more important. In the planning stages it is critical to identify mission requirements, determine both the power source and energy storage, and select means of regulation and control.

9.1 Requirements

9.1.1 Mission

What really defines the power system for a robot is its mission. One would not likely attach a nuclear reactor to an radio-controlled car or power the city of Pittsburgh with a hand-operated generator. Obviously these mismatch pairs are ludicrous but it is important realize that the power system must reasonably meet the requirements of the robot so the robot can perform its intended duty. By defining the types of mission Daedulus will to perform in a specific environment we can better design the power system.

In its first mission, Daedulus will explore terrestrial terrain as similar to that of the moon as possible. Such proposed testing areas include El Mal Pais, NM and Death Valley. Daedulus would ideally run for 15 days straight at 8 hrs a day while only resting at night. The primary goal here will be to test the mechanism itself while science is secondary

In the space-bound mission, there will be a significant differences other than the several orders of magnitude of difference in cost. Science will be the highlight since no one would pay to just see a robot walk on the moon. The complexity in the power system will be much greater. The power system must completely support the robot since there will no one there to swap batteries or clean the solar arrays if they get dirty from dust. Also, there is a greater limitation on what technologies that can be used since newer technologies such as batteries have yet to be proven in space.

9.1.2 System

Given the complexity of this robot, it would facilitate matters by viewing power usage on a system by system bases. The systems that characterize Daedalus are the mechanism, controller, perception, planning, science and communications.

The power usage for the physical mechanism is from motors and servo amplifiers involved in the lifting and striding motion of Daedulus. The amount of power required for a full vertical lift of the robots body against gravity, the assumed maximum power needed for any portion of a stroke, is used as the main component of mechanism usage. The other component is the amplifier loss that is associated with having the amplifiers on while the motors are not being used. Unlike the motors, the amplifiers must always be on.

The perception system consists of a Sparc IPX computer with two frame grabber boards for the requisite number of video cameras. As with much of the computing on Daedulus, the perception computer along with any associated boards must always be active at all times. The cameras

and its mounting hardware, on the other hand can be turned off when not in use as in case of performing science.

The controller system is contains several boards to process and communicate data from various other parts of the robot. Since this system controls everything it must be on at all times. Therefore both dedicated processor boards, the ethernet board and various relay, I/O, D/A and A./D cards must be on.

Planning is made up of another Sparc IPX computer. It too must be on at all times. Communications, currently consisting of a wireless ethernet connection, must also be active at all times.

Depending on the actual science package used, it may or ma not be required to on at all times. It is assumed that it will only need power when science occurs. Regardless, it is allotted some power.

Note, some components require cooling and are thus noted in the power budget table.

Daedalus Power Budget

System	Item	# of	Duty	Cooling	Fixed	Variable
Mechanism	Physical Lift Power		Variable	No		80.4 W
	Amplifier Losses		Variable	Yes	4 W	4 W
Controller	Processor Board	2	Fixed	Yes	9 W	
	Ethernet Board	1	Fixed	Yes	5 W	
	Digital I/O Board	2	Fixed	Yes	2 W	
	A/D Board	2	Fixed	Yes	2 W	
	D/A Board	1	Fixed	Yes	3.5 W	
	Relay Board	1	Fixed	Yes	3 W	
Perception	Sparc IPX	1	Fixed	Yes	40 W	
	Frame Grabber	2	Fixed	Yes	40 W	
	Camera	3	Fixed	No	18 W	
	Video Switch	1	Variable	No		10 W
	Camera Pan	1	Variable	No		10 W
Planning	Sparc IPX	1	Fixed	Yes	40 W	
Science	???	1	Variable	Yes		50 W
Communications	Wireless Ethernet	1	Fixed	Yes	18 W	

Power Budget Summary

System	Fixed	Variable
Mechanism		84.4 W
Controller	24.5	
Perception	98 W	20 W
Planning	40 W	
Science		50 W
Cooling	83.3 W	
Power Supply	12.3 W	7.7 W
Total	258.0 W	162.1 W

By combining the fixed and variable power usage in their respective system we determine the power usage of each system when it is fully active and when it is not. Factors such as cooling and power supply have been added as to impact overall consumption. This information can be used to build profiles of power usage for a specific mission.

9.1.3 Power Generation

Current plans for the earth-based mission will not include any source of power generation. All power for Daedulus will be derived from rechargeable batteries. After each day of operation, the used batteries will be replaced by a fresh ones.

For the lunar mission the only feasible system of power generation is the photovoltaic solar cells. Solar cells offer unlimited energy during daytime but is limited to total available surface area. Since one lunar day is equivalent of 15 earth days, there will not be a lack of light. Difficulties lie in the fact that solar cell performance varies with temperature, age and the incidence angle between the light and the cell. Since there is no intention to survive a lunar night, such large temperature fluctuations should not have an impact, unless the robot does walk into heavily shaded area. Performance degradation are usually measured in terms of month or years. Since we are dealing with a couple of days, we can ignore age. The main problem is primarily incidence angle and surface area.

Solar cells would be mounted directly onto the robot's tapered hexagon shaped upper body. The total amount of area available has been estimated to be about 1 sq-m. From a sample of solar panel currently available, we see that for a given area assuming that the whole array receives sunlight at an angle perpendicular to the panel that the total power available is 172 W at 168 W/kg.

$$\begin{aligned}
 \text{Power to Weight} &= (\text{Power per sq. ft.})/(\text{Weight per sq. ft.}) \\
 &= (16 \text{ W/sq ft.})/(0.095 \text{ kg/sq ft}) \\
 &= 168 \text{ W/kg}
 \end{aligned}$$

$$\begin{aligned}
 \text{Surface Area Available} &= 1 \text{ sq m} \\
 \text{Power} &= (16 \text{ W/sq ft}) \cdot (10.76 \text{ sq ft/sq m}) \cdot 1 \text{ m} \\
 &= 172 \text{ W}
 \end{aligned}$$

From these calculations, it is evident that it will be difficult to get enough power without adding too much weight to the robot. This is of great importance since the solar cells will be the only means of power generation. Other than selecting solar cells with better performance characteristics, the amount of surface area might need to be increased. Multi-hinged solar cells that unfold when the robot lands and tilts towards light as the robot move might increase available power at the cost of complexity.

9.2 Energy Storage

For the earth bound mission, the main power source for Daedulus will be secondary batteries. Running time has been chosen to be 8-10 hours of operation. When these batteries are exhausted they will be replaced by technicality. On the lunar mission, solar cells will be used to charge secondary batteries. also UPS will be implemented to clean the power.

On the moon, Daedulus will perform its mission for the full duration in daylight, 8 hrs at a time, taking breaks only to have the batteries to be recharged by the solar cells. Another possibility is to allow for multiple batteries: one set of batteries will be used while the solar array will charge one or more sets of batteries. This way, Daedulus can operate continuously for a full lunar day if the robot's operators so choose so.

Currently, there is a wealth of secondary battery technologies but the choice of battery technology has been narrowed down to metal hydride and nickel zinc. The reasons are that both technologies provide superior power/weight and weight/volume ratios, have been proven for robotic applications and have a lifetime of about 600 to 1000 cycles. Below is a summary of various technologies with relative characteristics.

Ni-MH

Power to Weight = 180 W/kg
 Energy to Weight = 60 - 70 W•hr/kg

NiZn

Power to Weight = 180 W/kg
 Energy to Weight = 58 - 65 W•hr/kg

NiCad

Power to Weight = 180 W/kg
 Energy to Weight = 25 - 40 W•hr/kg

Lead Acid

Power to Weight = 100 W/kg
 Energy to Weight = 25 -30 W•hr/kg

9.3 Regulation and Control

No details on regulation and control for management is included in this document since it requires better definition of the robot itself which is currently not available. It must also be noted that a UPS in conjunction with a power generation and energy storage system must be used. This element is

crucial to the proper operation of the robot since unclean power can be the source of many problems.

9.4 Sample Mission

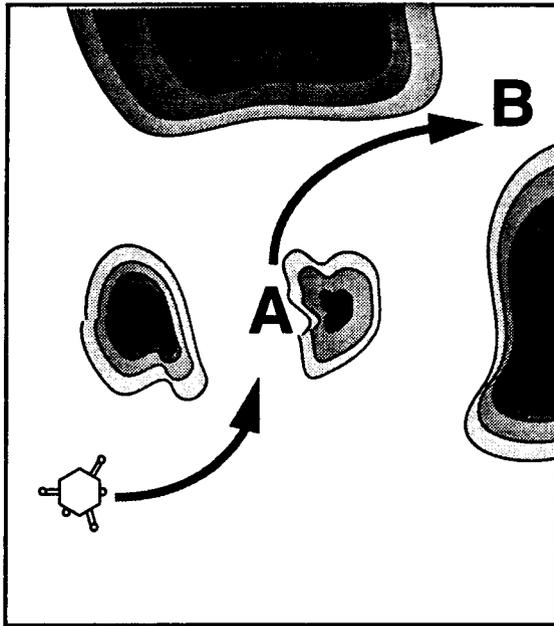
To better understand the characteristics of the power management requirements it is good to have an illustrative but hypothetical example. It is the first day of a 15 day mission in El Mal, NM. Daedulus has been off loaded to a test sight sometime in the morning. It is fully charged and armed with its trusty soil analyzer as its sole scientific instrument. It has determined for itself that the best course of action is to wonder around the area and perform several soil tests.

For the first several hours, Daedulus wonders around the area until it picks a sight that would be best suited to take a soil sample. On the forth hour, the robot stops movement, picks a soil sample. Since processing of the soil takes one hour and all accuracy is lost with the robot moving while processing occurs, the robot remains stationary until the fifth hour which it continues roaming until it finds another suitable sight two hours later. Again, the same restriction apply so Daedulus remains stationary for the duration of soil processing. By now it is very late and there no light for the robot to see so it decides to power down and sleep until the next day. Given the physical mechanisms speed of 10 m/min, Daedulus might have traveled close to 2.5 miles and has gathered a very minimal amount of data about the quantitative nature of the soil in the area occupied by the robot.

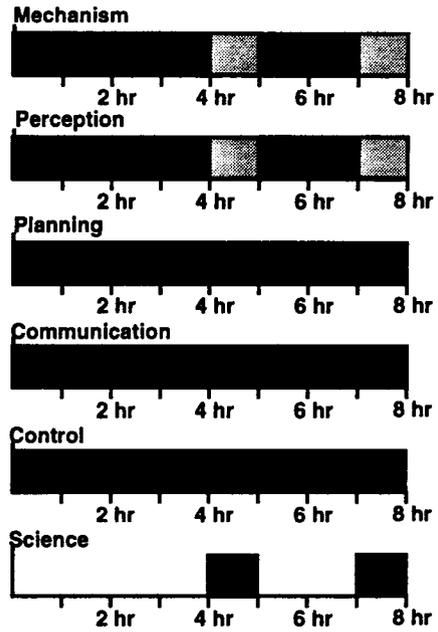
From the included chart we see there are systems that are on running on drawing peak, minimal and no power which are denoted by the dark, light and clear regions respectively. Planning, perception and communications subsystems are active as long as the robot is and uses constant power. On the other hand, science is only active when it is used else it uses no power at all. When the mechanism is walking both mechanism and perception systems are active. When the robot is not moving, such as when science occurs, these subsystems are using minimal power. Some reasons why these systems are at minimal instead of off include motor amplifiers must be active at all times even Daedulus is stationary and perception might need its cameras on to gather and process information.

A power consumption versus time shows the overall power consumption of the robot for a given time through the mission. From the previous tables we see that the robot consumes approximately 260 W for the 6 hours that it travelling and 160 W for the 2 hours it took the robot to analyze the soil thus requiring a total of 1880 W-hr.

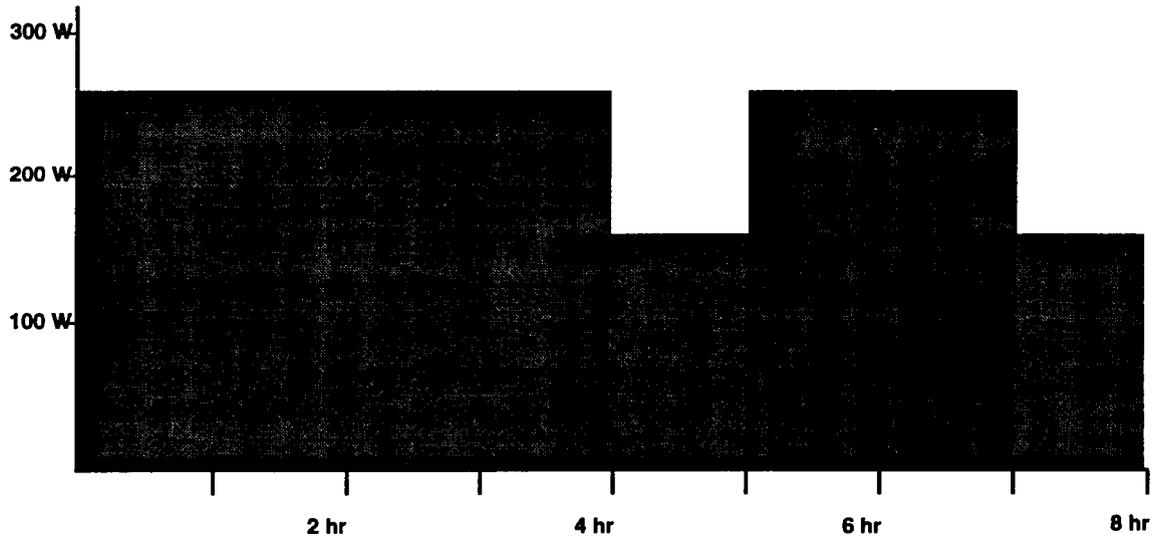
Sample Mission



Medium Range with Some Science



System Activity During Mission



Power Consumption versus Mission Time

$$C = \frac{P \times T}{D \times N \times V \times n}$$

T= running time= 8 hrs
D= limit on depth of discharge= 80%
N= number of batteries= 1
V= average discharge voltage= 24 volts
n= transmission efficiency= 90%
C= rated battery capacity=194 A-hrs

In specific, the current performance characteristics of Ni-MH technology has the specific power of 200 W/kg and the energy density of 56 W*hr/kg. Given the requirements of our sample mission the battery weight is about 34kg while offering a total available power of 6,800 W.

energy= 260 W 6 hrs + 160 W* 2 hr= 1880 W*hr*
*battery weight= 1880 W*hr/ 56 W*hr/kg= 34 kg*
total available power= 200 W/kg 34 kg= 6,800 W*

Energy storage capacity is determined to be roughly 194 A-hrs. This is derived by running Daedulus for a period of 8hrs on one battery with a hypothetically assumed transmission from battery to system efficiency of 90% for a a hypothetically limit depth of discharge for a battery of 80% with a bus voltage of 24 volts. The A-hrs rating is possible since current battery technology used for electric cars can be used to meet the necessary requirements.

9.5 Conclusion

Using ball park numbers for the power and mass budgets, it is possible to solely use batteries for a earth mission with the use of battery swaps. For the space mission, due to the limitations of power generation power management will be much more difficult aside from the general issues involved in making a system space worthy.

Another issue worth investigating is to determine whether is desirable for Daedulus function round the clock on a lunar mission. The original intention was for Daedulus to run 8 hrs at a time. It would perform some tasks for about 8 hrs and then shut down. When it is down the solar panels would re-charge the batteries. Upon completion of the recharge, Daedulus would then run for another 8 hrs which it then continues the cycle. The new situation would involve Daedulus running on a set of batteries while the second set of batteries are recharged. After a 8 hr period, the robot will switch over to the second set of batteries while the solar panel recharges the first set that has just been expended.

10 Software

10.1 Software design

For a complex piece of software to succeed, it must be organized in a coherent form. When a given form is non-changing, it is referred to as an architecture. This architecture serves as the backbone upon which a software system is constructed. A well devised architecture provides robustness, facilitates development, ensures reliability and predictability, and guarantees performance. More specifically, when dealing with an architecture suitable for the control of robotic systems, the following capabilities are desirable:

- A robot “operating system”
- Support for performing tasks, planning and execution
- Support for reacting to changes, monitoring, concurrency and error recovery
- Conflict resolution, prioritizing goals/commands, sensor fusion and resource management.

The capabilities must be provided in face of

- A complex, dynamic, unstructured environment
- Complex tasks with competing deadlines
- Limited sensors and computational resources

Existing computer architectures can be categorized into one of three types:

- **Hierarchy** - Tasks are decomposed based on given criteria, including, behavioral, functional, temporal and spatial. Hierarchical architectures are applicable for planning, perception, error recovery strategies, etc.
- **Reactive** - The system is driven by inputs from the environment, without advanced planning. Sensors are continually monitored, decision cycles are short and there is decentralized control. The architectures work well on simple tasks.
- **Deliberative** - The task cycle is comprised of three steps: plan, monitor and replan. These architectures permit multiple focuses of attention, for example, selective monitoring, prioritization of goal and resource management. The system are scalable and facilitate goal planning, however, there is typically no guaranteed response time and there exists possible centralized bottlenecks.

A computer architecture for a real system might comprise several of these concepts within a single software system. The overall architecture might be deliberative, certain modules would be hierarchical and the real time controller reactive. It is this type of hybrid system that is proposed.

10.2 Architectures

10.2.1 Task Control Architecture

A flexible, powerful software architecture is needed to coordinate tasks and control complex robotic systems. The Task Control Architecture (TCA) is designed specifically for robot systems that must operate autonomously for extended periods of time in uncertain, dynamic, and rich environments. In addition, the TCA is designed for robots that have multiple tasks to achieve, but limited sensors and computational resources relative to their tasks. The TCA provides a vocabulary of high-level constructs for specifying the component interactions in distributed robot systems. It also provides software utilities that implement the constructs. A system built using TCA consists of a number of distributed modules (processes) and a task-independent central control module. The proposed architecture, with associated packet size and frequency, is depicted in Figure 10.2-1 and Figure 10.2-2.

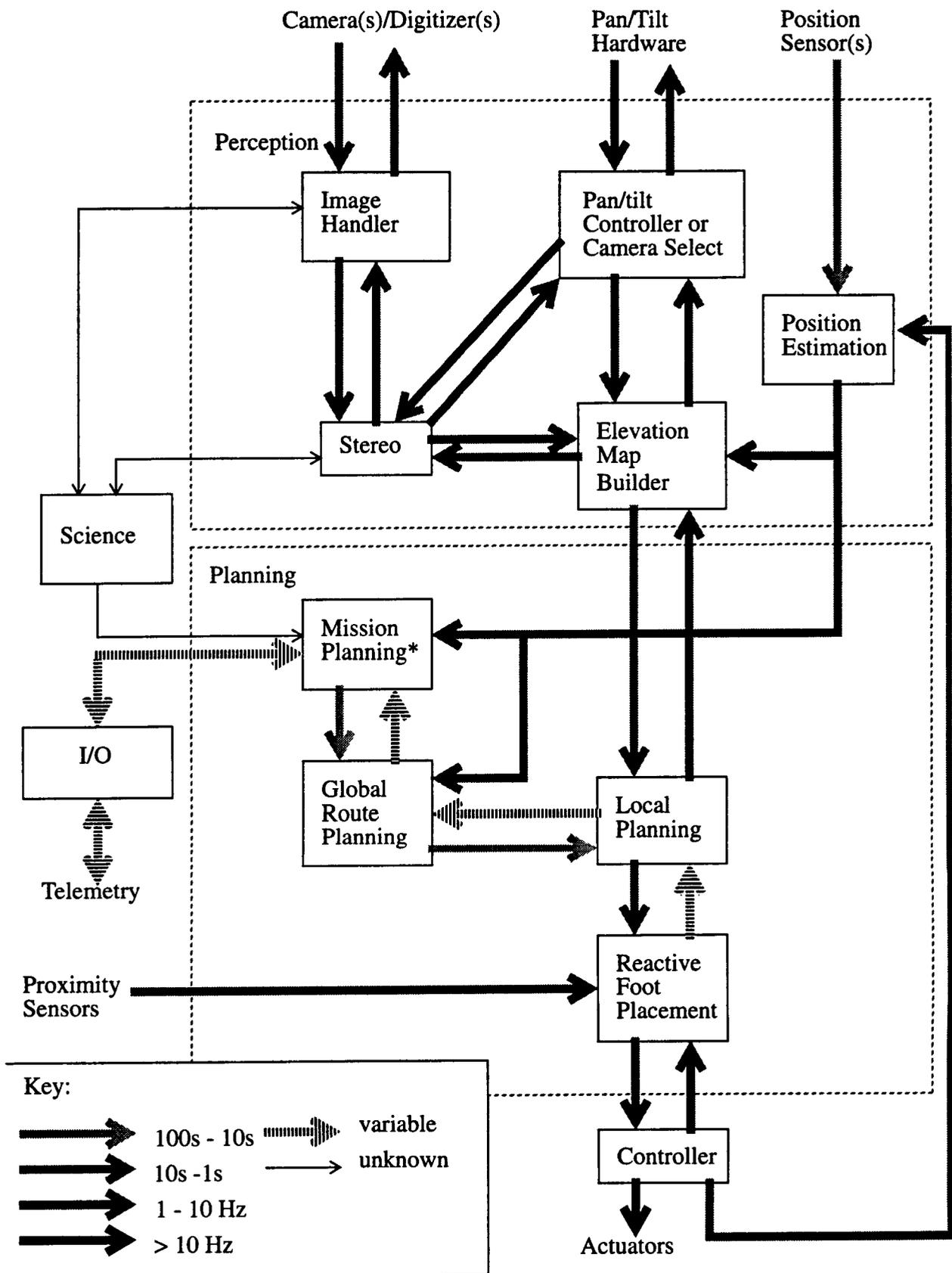


Figure 10.2-1 Packet Frequency

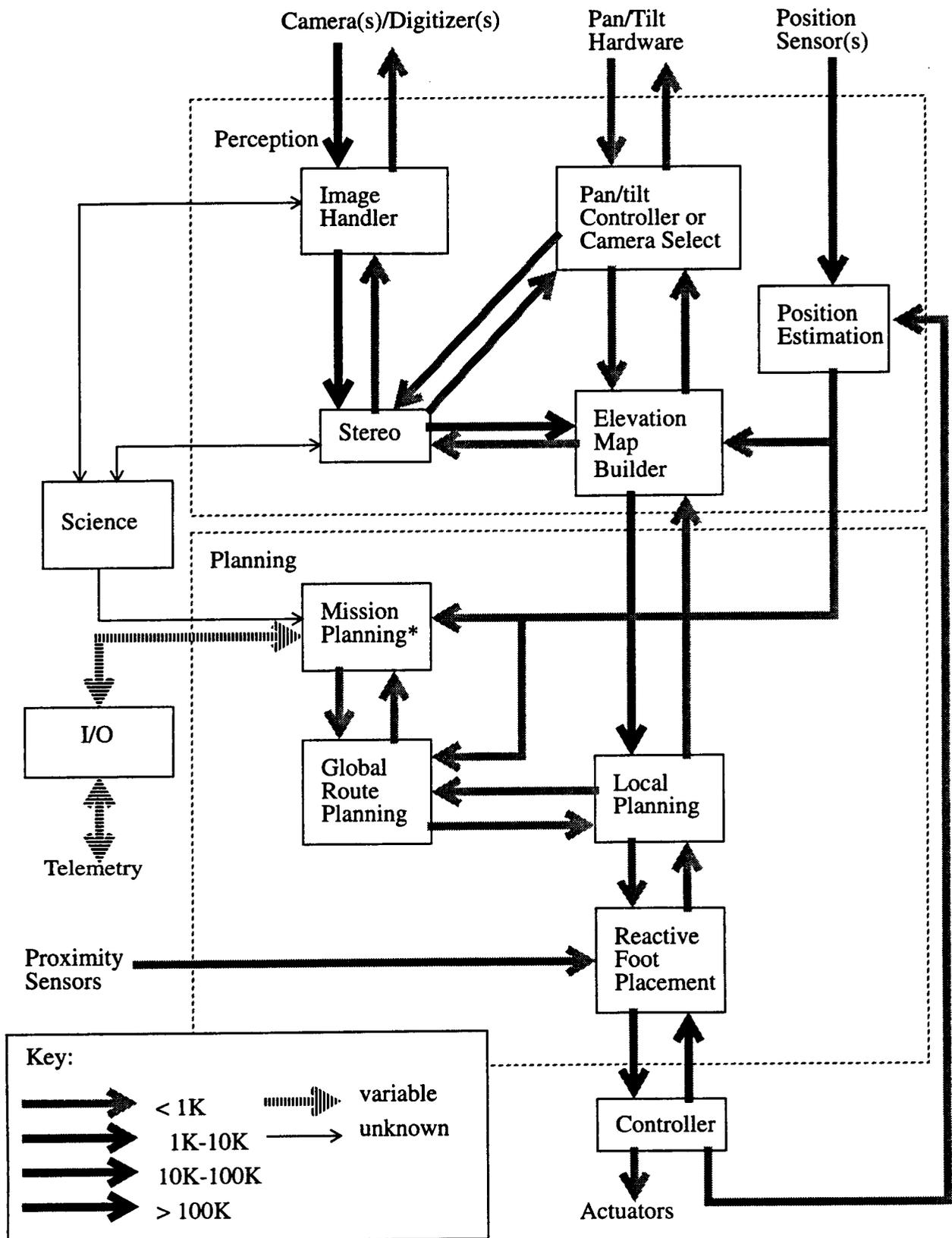


Figure 10.2-2 Packet Size

10.3 Architecture modules

The software is arranged in modules, each of which serves a particular function. Using the TCA, the modules are not restricted to any particular computer platform. An additional benefit of this architecture is that modules can be easily replaced as needed.

Most Daedalus software modules can be classified into one of two categories: perception or planning. There are a few modules that do not fall into either category, such as the science package for dealing with any interesting lunar geology the robot may find. The modules in the perception subsystem will communicate with other modules on demand; that is, when the module receives a request from another module, it will supply the requested information as soon as possible. If, for some reason, no information is available from which to satisfy the request, a failure may be returned. In contrast, the modules in the planning subsystem will send new information to other modules whenever that information is available.

10.3.1 Perception Modules

10.3.1.1 The Image Handler

The purpose of the image handler is to provide a software interface for the imaging hardware, as well as some time history capability. Since multiple modules may wish to obtain images of the environment surrounding the robot, it is necessary to have a software module to act as an arbitrator of the hardware.

10.3.1.2 Pan/Tilt Controller or Camera Selector

Since it may be necessary for the robot to look in directions other than straight ahead in order to navigate or perform science tasks on the moon, it will be necessary to have either multiple cameras or a pan/tilt mount for a single camera. Given that multi-baseline stereo is currently planned for the vehicle, multiple cameras will definitely be present. Since only one digitizer board is currently planned, signal selection hardware will be needed.

Again, since multiple modules may wish to manipulate the hardware, this module will provide a common interface as well as command arbitration. The module would respond to two types of requests: another module can either query the current state of the hardware or request a change in that state. It might be possible to merge this module with the image handler with no degradation of performance.

10.3.1.3 Stereo

The purpose of this module is to compute stereo range data from individual video images. Another module may request a range map of a specific area in the local environment of the robot, at a specified resolution, and this module will respond with the requested range image, if possible.

10.3.1.4 Elevation Map Builder

It is the job of this module to take stereo range data and convert it to a local elevation map. This module is responsible for responding to requests for an elevation map of a given area of the surrounding terrain. This may require multiple maps to be retained and merged in order to provide a map, for instance, of the area beneath the robot.

Other modules can request maps of specific locations at a variety of desired resolutions, and the Elevation Map Builder will respond with the best possible map given the current situation. Note that such requests may fail due to lack of data.

10.3.1.5 Position Estimation

This module will provide an estimate of the global position of the robot (in global coordinates), along with an uncertainty measure on that estimate, to any module that requests it. This module is

responsible for merging position estimates from a variety of different sources, including position sensors and dead reckoning information from the vehicle controller.

10.3.2 Planning Modules

10.3.2.1 Mission Planner

The mission planner is responsible for locating regions of interest, and setting high-level goals for the Daedalus robot. Given the global position of the robot, it will locate “interesting” regions nearby, and prioritize them by how interesting they are and an estimate of the amount of time that would have to be devoted to that goal. It will then choose the best of these possible goals as the next goal for the robot to pursue.

This information will then be passed down to the global planner. If some new piece of information from the sensors or ground control changes the goals of the robot at any time, the new goal can be passed to the global route planner and propagated down through the system. If, for some reason, the lower level planners are not able to achieve the stated goal, they will fail and return to the global planner, which will choose the next best goal.

10.3.2.2 Global Route Planner

The global route planner will take the stated goals from the mission planner, and plan a global path from the current position to the goal. This planner will probably make use of an a priori global elevation map in order to determine the best possible path.

A direction of travel will then be passed on to the local planner. The local planner will then attempt to make headway in the specified direction. As the robot moves, the global planner will continue to indicate the optimal direction of travel. Since it is unlikely that the robot will be able to follow the path as it is initially planned, the global planner needs to be able to indicate an optimal direction from any conceivable robot position.

If for some reason (a large wall-like obstacle, for instance) the local planner finds itself completely unable to make progress in the indicated direction, it will report a failure, and the global planner will then choose an alternate direction.

10.3.2.3 Local Planner

The local planner operates in one of two modes: reactive and deliberative.

In reactive mode, the local planner accepts desired direction information from the global planner, and attempts to guide the robot through the local terrain while making progress in that direction. The local planner will request local elevation maps from the elevation map builder. By determining which regions of the map are admissible, the local planner will then direct the reactive foot placement module as to how to avoid the inadmissible regions.

In deliberative mode, the local planner will make use of very high resolution elevation maps of the surrounding terrain. It will then select individual foot placements for the reactive foot placement algorithm. This mode will only be useful in the roughest terrain that the robot can handle.

10.3.2.4 Reactive Foot Placement

The reactive foot placement module takes either a general direction or more specific instructions from the local planner, and communicates directly with the vehicle controller. Using inputs from proximity and contact sensors mounted on the feet, it will pick and choose foot placements, and handle gait generation.

10.3.3 Other Modules

10.3.3.1 Controller

For reasons similar to those listed in Section 10.3.1, the controller module acts as the interface between the planning modules, that command motions, and the actual motion control hardware. By providing this interface, the user can specify desired motions and not be concerned with the implementation details for the mechanical system.

10.3.3.2 I/O module

The I/O module is provided so that the human operator, who is typically not colocated with the Daedalus, may control all aspects of the rovers activities. It provides an interface between the remote human operator and all of the software and hardware on the robot. This interface is maintained at all times by direct communication through the telemetry hardware. Typically, the operator would provide high-level commands to the system and monitor its progress. However, under certain circumstances, it may be necessary for the operator to take more direct control of the Daedalus walker.

10.3.3.3 Data Logger

For debugging purposes, it will be necessary to log data from a variety of modules in order to track their progress and isolate the reasons for their failure. This facility can be provided by a centralized data logging server. It would then be possible to replay the situation in simulation and locate the cause of the failure.

10.3.3.4 Science Module(s)

Finally, the scientific nature of this mission will require a suite of scientific sensors, with an associated module or modules to control them and direct the progress of the mission.

10.4 Control

The Daedalus robot physical controller is a successor of the Ambler controller. Fundamental controller architecture is similar to its forefather, but we have to consider some differences:

- Daedalus is a frame walker while the Ambler is a circulating-gait walker.
- Daedalus has to be less expensive and smaller.
- Daedalus has to be almost ten times faster in overall locomotion (10 m/min).

While the mechanically decoupled leg configuration of the Daedalus reduce the computational cost for generation of motion control references, foot placement planning becomes difficult because of multiple feet placement. For faster locomotion, reactive foot placement planner is to be added in the controller. We also discuss the respose improvement of the body tilt recovery.

10.4.1 Physical (low-level) control

As stated above, total computational demand is as same as at the Ambler, hardware composition is two VME board (68020 or 68030 base) for physical control and I/Os including servo motor amplifiers for actuators. VxWorks is the most likely OS of this system like other mobile robots in FRC in CMU. Control schemes will be written in C language. This is already discussed at section 6.2.

10.4.1.1 Functions of physical controller

Tasks for the physical controller to be performed are:

- To generate motion by commanding each actuator

One of them is to decide actual foot placement in the freedom of stride/traverse and rotation angle of frame/body to move. Another is to generate gate (or body motion). One typical gate is a simultaneous actuation that perform to speed up the

locomotion in order to fulfill the premise at a sacrifice of larger power consumption. Other is sequential actuation where only one actuator is activated at a time.

- To level the body of robot
- To monitor the health condition of sensors and actuators
- To report the internal sensor data

In order to specify the role of the physical controller, nominal I/O is considered. The physical controller is an module in total control system, and has both interface to environment and other module of system.

About the input from sensors, vertical leg force (or contact) sensor input from each leg is required. While the Ambler can refer the quantitative information of force exerted on each legs, the Daedalus is provided only by bounded force sensing, contact or tactile information due to its weight and budget limitation. For foot placement behavior, close proximity (or short range distance) sensor for each leg have to be provided. This sensing can be omnidirectional or small field of view (FOV) in forward direction.

Another input is of communication from the other modules. Reactive planning module in controller refers elevation maps. It is quite probable that the environments of the robot can change quickly, therefore the map has to be selectable between coarse and fine map, or body size resolution map and foot size resolution map depends on the confronting environment. Frequent change of body height can cause unnecessary energy consumption, so stable body height reference is important.

The most significant output from the physical controller is for actuators. Input signals for them is motor position command because of the servo motor amplifier, and also it is supposed to perform stepwise or trapezoidal velocity profiles.

The other required output is for communication. Physical controller should inform the abnormal status of major mechanical component like motor, motor amplifier, sensor and physical controller itself including the physical controller software. The data for dead reckoning and body attitude and height should be calculated and sent to the perception unit as an internal sensor report.

10.4.1.2 Architecture of physical controller

The physical controller consists from four major units:

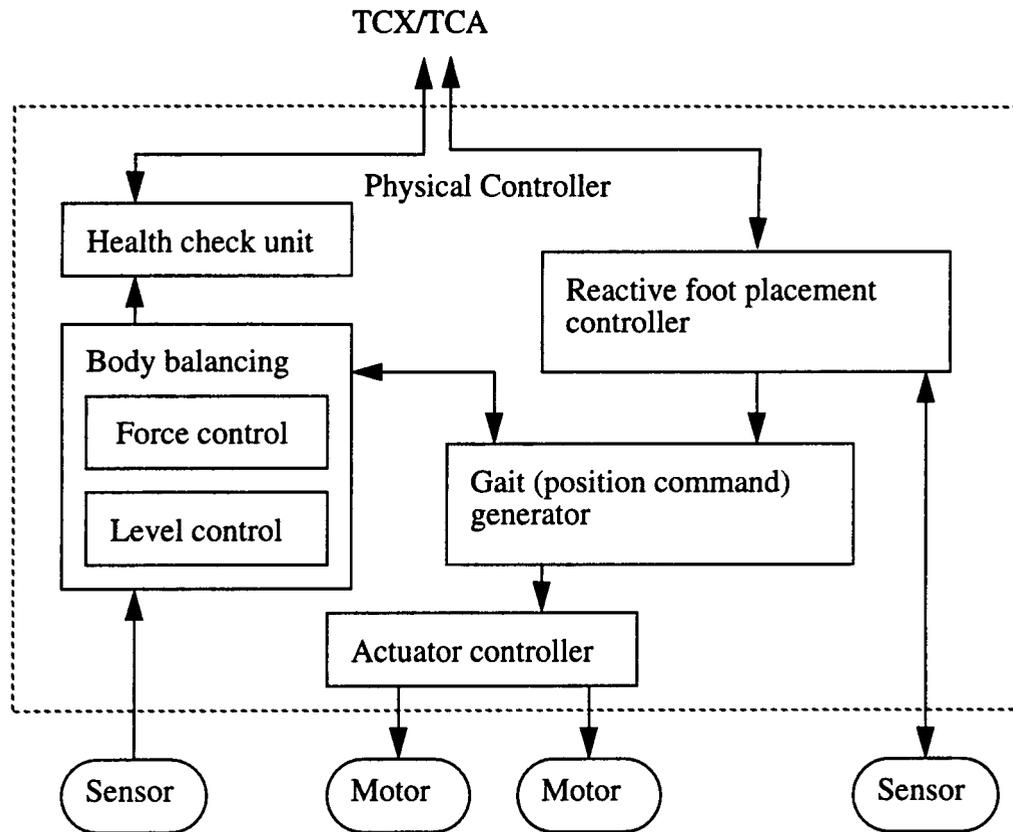


Figure 10.4-1 Architecture of physical controller

Reactive foot placement unit will be discussed in section 10.4.2 in detail.

- Health check unit

There is no significant change form the physical controller of the Ambler. The cycle of checking is minimum unit of clock.

- Gait generator

This unit generates the next referring point in each clock cycle (for instance 2~4msec) in both sequential and simultaneous mode and linked with several inter-ruption connection form body balancing unit and health check unit. The cycle of commanding is minimum unit of clock.

- Body balancing unit

This unit receives force/contact sensor input and level sensor input and gives halt command or corrective motion related to z-axis to the gate generator.

10.4.1.3 Body balancing

While both the Daedalus and the Ambler have six legs, the Ambler rewrites only one leg at a chance and three legs keep on supporting its weight in addition to rest of two legs which are served for better force distribution. Since the Daedalus is a frame walker, it is always supported only three legs, hence the failure of the leg support is crucial. This is a motivation of implementing more elaborate balancing scheme than the Ambler has. In order to avoid the hunting of the body,

deadband for input signal of force/contact sensor and inclinometer is employed. Inclinometer input deadband is around 0.1 degree of incline. The force sensor data is checked in the shortest cycle where they exceeds certain boundary quantity. Since the non-linearity is nature of the force/contact sensor corresponding action should be mild. The inclinometer input is checked in the leveling behavior control unit. It interprets the quantity of the body tilt as failure of the support of the leg if the tilt exceeds predetermined threshold, for example around ten degree. Otherwise it regard the tilt as accumulated error of walking steps. Deadband is required to prevent hunting of the body as the Ambler experienced using the deadband.

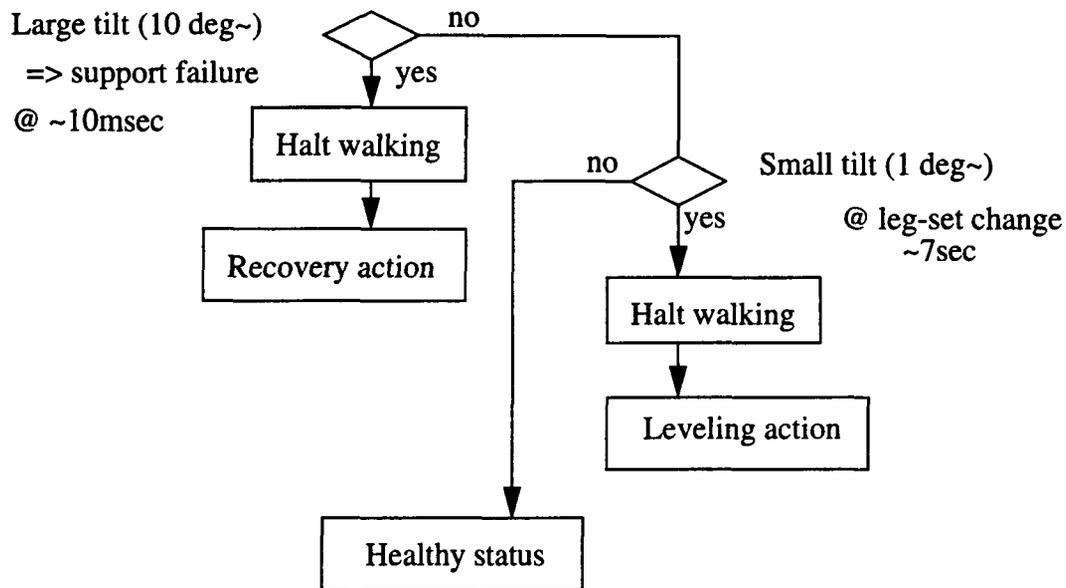


Figure 10.4-2 Leveling behavior control

Fig.10.4-2 shows the scheme of leveling control. Large tilt is checked in minimum cycle time and corresponding action is failure recovery, while small tilt is seen in each step of walking motion (or each exchange of leg set) and leveling action is taken.

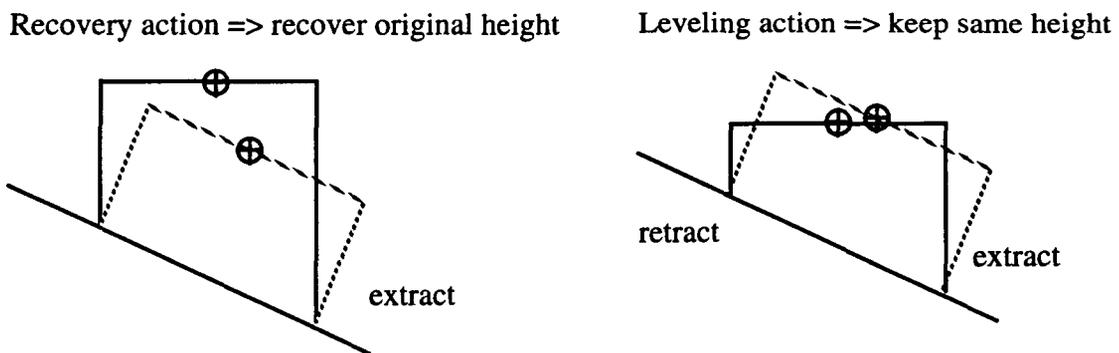


Figure 10.4-3 Recovery and leveling

Fig10.4-3 shows the failure recovery action and leveling action. Since recovery action is supposing the support failure, idea of this action is to recover the original height of the body. On the other hand leveling action assumes that least change of body height is most likely for correcting action. Only the leveling action was implemented in the Ambler.

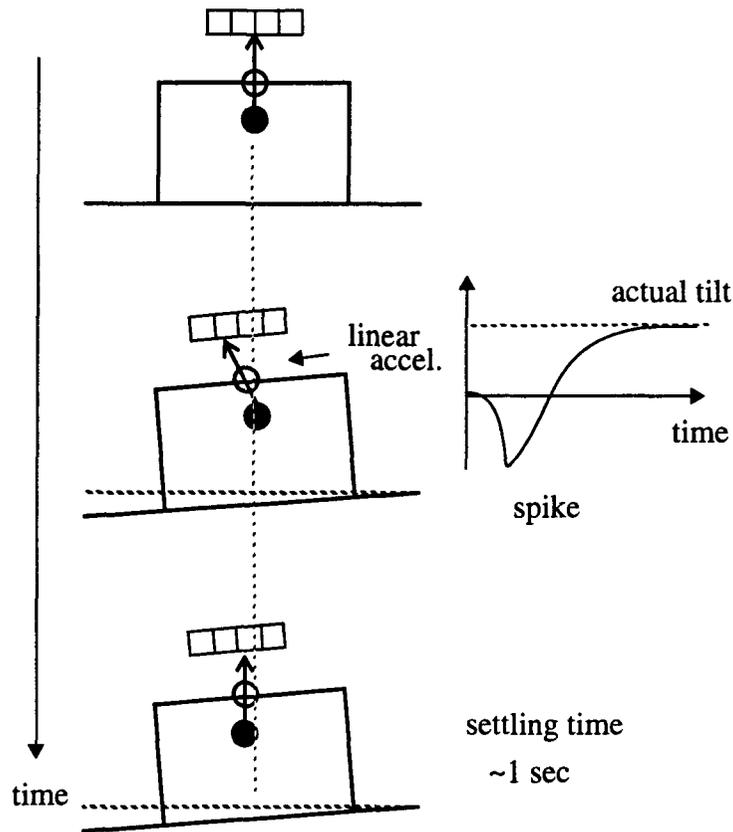


Figure 10.4-4 Function of the inclinometer

Fig. 10.4-4 shows the characteristic of ordinary (mechanical) inclinometer. We can assume that the mechanical model of the inclinometer as a simple pendulum with viscous friction model. Since it is sensible to the linear acceleration of the supporting point (body or frame) and has long time constant (order of ~ 1 sec), we should consider the sampling timing in the walking motion for settling time of inclinometer. At the Ambler, the sampling interval for the inclinometer is longer than one second¹ because of the overdamping character of inclinometer pendulum for stability. In order to overcome this sensor response problem, we can use the inclinometer dynamics for estimating the body tilt through the state variables such as inclinometer deflection angle, velocity of the angle, and the acceleration of the angle. Fig.10.4-5 shows this scheme.

1. In Nagy, Peter G. *An investigation of walker/terrain interaction*. Thesis(Ph.D) Mechanical Engineering Department, Carnegie Mellon University, 1991

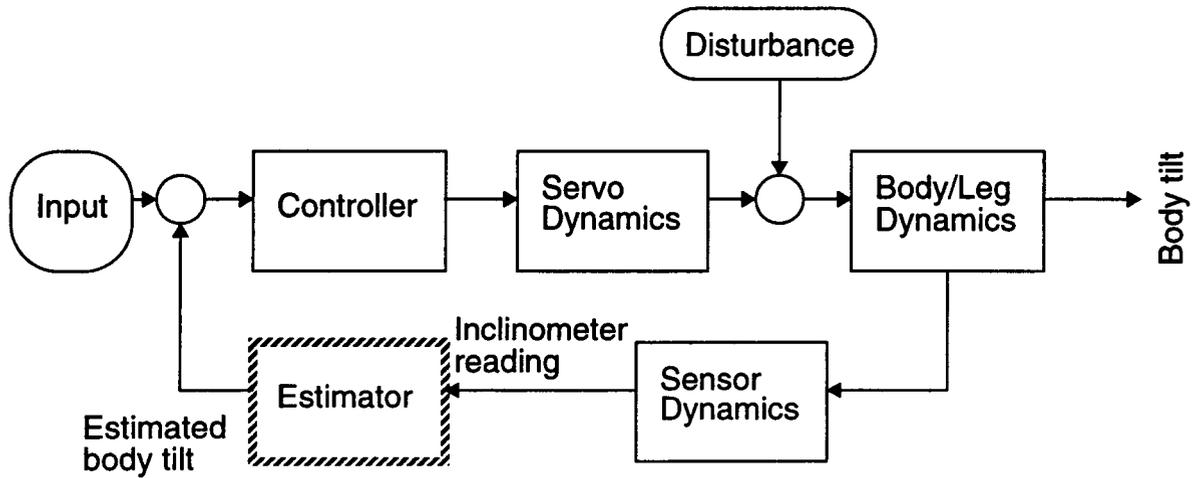


Figure 10.4-5 Control system with estimator

Though it is difficult to measure the state variables directly, we can substitute the difference value instead of the time differential value, or velocity, of the deflection angle of the inclinometer pendulum. Similarly for acceleration. Alternative is a observer of the plant physics and this approach is a proven method in state-space controller design.

We can show a simple simulation that indicates the effectiveness of this estimator. Suppose a mechanical system shown in Fig.10.4-6. Assumptions are 'frame is rigid', 'remaining (left) foot does not move', and 'pendulum mass is small'. Dynamic equation of motion is

$$m\ddot{r}_m = Tc_2 + mg - d[\dot{r}_m - (\dot{r}_p + \phi L_p c_1)]$$

<Dynamic equation> 10.4-1

Where, r_m is position vector of mass of pendulum, r_p is position vector of supporting point of the pendulum, d is damping factor of pendulum, ϕ is the angle of body tilt, and θ is the angle of pendulum.

equation 10.4-3. This is a estimating equation for φ only by θ 's. This means continuous measuring of θ can tell the estimation of φ before the inclinometer settle down. This linearization is quite apart from the actual physical meaning, but the purpose of this linearization might be interpreted as first order approximation.

Figure 10.4-7 Typical result of body tilt estimator at numerical simulation

Fig.10.4-7 shows a typical sample of numerical simulation of this estimation. Solid line is a actual φ value (or the angle of the body tilt) while estimated φ value is shown as dashed line. In this simulation we assumed that the body falls freely until the tilt become around ten degree. Pendulum system as an inclinometer has overdamping character because of its around one second of settling time. The settling time for the critical damping of this pendulum is around 0.2 second, therefore we can regard that this pendulum has a similar characteristic of actual pendulum. From the Fig.10.4-7, estimated value of φ is quite close to the final value of actual φ . And we can use this scheme 0.1 second later the leg on the support failure get contact with ground. The possible reason of this incorrectness of estimation while body is free falling may be the assumption that $\dot{\varphi}=0$ is applicable for local linearization, since the angular acceleration can be large at that moment. The equation 10.4-3 is considerably inexpensive in term of the control computer load. We can approximate first and second derivatives of θ by first and second difference of each sampling θ . Hence this estimator can improve the level action response with small cost. We have to conduct actual experiment to prove its effectiveness before installing.

Other practical but most fundamental improvement is faster sensor. For inclinometer, there are several principals including mechanical phenomena. We have a magnetic inclinometer whose response speed is around 30Hz while its price is around \$600 for single axis. If this sensor is valid for Daedalus, this sensor is most likely solution for improvement of leveling control speed up. In this case body tilt estimator described previously has no significant meaning, because the bottle neck of tilt recovery response becomes speed limitation of leg actuator, which is 0.25 meter vertical motion per 8.3 second.

As a conclusion for body balancing control, while most of scheme is same as of the Ambler, we propose the faster response scheme for the Daedalus. That is implementation of faster sensor (non-mechanical principle inclinometer), otherwise controller estimation method.

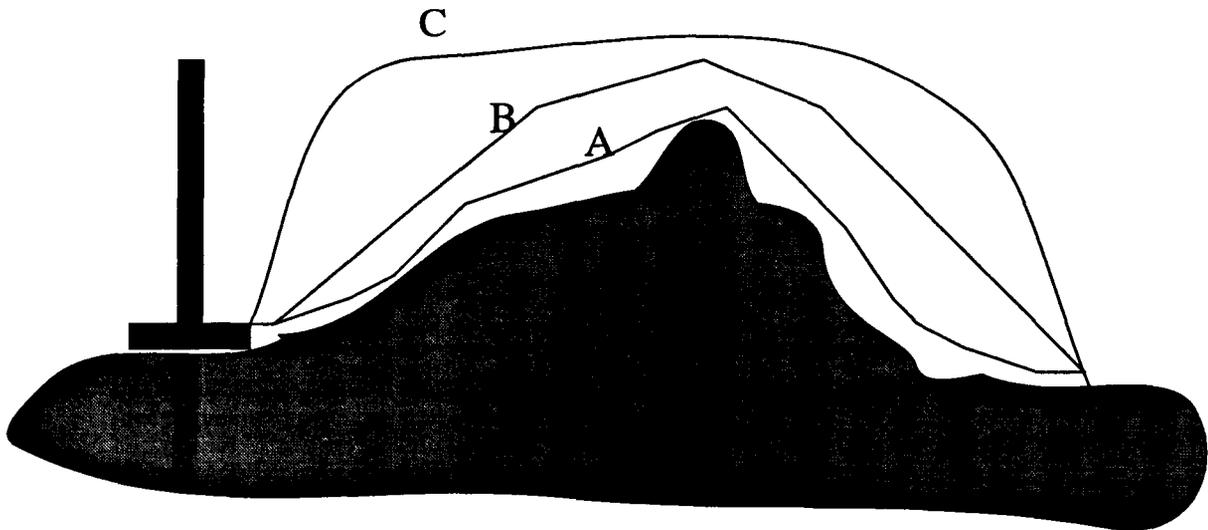
Additional proposal: since the Daedalus has fixed leg configuration in body and frame, I would like to propose a reactive leveling control scheme, which can be implemented outside the physical control computer. Locating the inclinometer on the diagonals of each leg, we can implement the leveling controller without computing the tilt angle. Considering the cost of the inclinometer is around \$200 per unit for one dimensional sensor and \$300 per unit for two-dimensional), increase of the physical cost is not significant.

10.4.2 Reactive gait planning and motion control

The primary objective of reactive foot placement planning is to use moderate computational resources to affect robust, effecient walking behavior. The final goal is to use the 608020/30 processor power and local sensors to walk as Fast As the System (FAS) permits. A recommended local sensor suite will be a byproduct of this work. The nominal, maximal walking speed assumed for analysis is 10m/min and the nominal walking step in 6.75s is

$$Step = \frac{10 \frac{m}{min}}{4.44 \frac{cycles}{min} \times 2 \frac{steps}{cycle}} = 1.13m$$

The figure below shows three distinct qualitative leg motions:



- First line of defense mode using rugged proximity/contact sensors. This mode is needed for fault tolerance and rugged behavior.
- Reactive leg and body motion based on distance sensors. The sensors here can look ahead to a limited extent and hence unplanned mid-course corrections may be needed.
- Ideal, based on stereo/range sensors. This is the fastest and safest method of walking, where the terrain covered in a single step is visible before motion and hence planned for.

The best sensor candidate for A is the capaciflector array discussed later(?). Mode B merges with C if the look-ahead distance is sufficient for complete recovery from any previously unknown obstacle. The velocity profile in Figure 3.1-4 gives the max stopping distance of about 0.25 m(check?). Assuming a body fixed sensor, the least sensor range for walking at max projected speed at full height (1.0L = 1m?) is 1.02m. The degradation of allowable speed and height with reduction in sensor range is given by:

$$\sqrt{H_{max}^2 + 0.04 \left(\frac{V^2}{V_{max}^2} \right)} \leq \frac{Range}{1.02}$$

For example, using sensors with a 0.5m range it is still possible to walk at max velocity as long as the body height is less than 0.44m. The ratio V/V_{max} gives the allowable velocity as a ratio of the mechanisms current capacity. This approximately translates to average speed as well. This equation will be translated to an interpolated lookup table for max speed versus height and sensor range, though the latter may eventually cease to be a variable. A similar analysis on feet proximity will yield velocity bounds on leg motions near the ground.

Walking efficiency will also depend on the ability to predict a suitable body lift. Stability considerations may require lower body lift, especially on slopes. Conversely, the allowable velocity is a function of body lift and slope due to stability considerations. The above equation places another on body lift and velocity. And finally, the highest known or expected obstacle gives the lowest possible body lift, since body motions to avoid obstacles are highly power expensive (3.32s/0.1m @ max power) and to be avoided. Ignoring obstacles, the body lift is obtained by optimizing for allowable velocity using stability and sensor range constraints. The obstacle height will override if this height is less than the expected obstacle height. The ability to learn or predict required body lift in a large obstacle terrain will be useful. A simplistic learning technique is:

$$H_{expected}[n] = \text{Max}(H_{observed}[i]); i = n - 10 \dots n - 1$$

The minimum body height can now be learnt as below. Using more sophisticated techniques like analysis of height profile rather than max height to generate expected height will improve optimality by reducing the standard deviation values in the equation below. A suggested technique is to generate a 99.9 percentile height correlated to the profile of the previous step to the current step. This is an attempt at a crude form of spectrographic analysis. Fractal based methods may prove to be better due to nature's tendency toward fractals.

$$Hmin [n] = Hexpected [n] \left\langle Mean \left(\frac{Hobserved [i]}{Hexpected [i]} \right) + 3 \times \sigma \left(\frac{Hobserved [i]}{Hexpected [i]} \right) \right\rangle$$

$$i = n - 100 \dots n - 1$$

Obstacle avoidance can be effected using evidence grids to fuse sensor readings to generate an obstacle map and to use Artificial Potential fields method or a modified Ganesha system to navigate through this map. Simulating walking in a terrain can help iron out bugs independent of the mechanism and converge to a subset of the techniques suggested above or invalidate them if unsuitable. In a cluttered field, the walking function will be taken over by the deliberative module which currently proposes to use stereo and long processing time to generate elaborate local terrain maps. Alternately, an elaborate local terrain map can be generating by using the frame itself as an active sensing platform. It is important to incorporate control delay into the simulator and test its effect on body dynamics. This will help in quantifying the computational power necessary for body and leg control.

In the deliberative mode, the final body and leg positions along with via points will be fed to the reactive module which will proceed normally, this time using the via points and specified positions as subgoals. It is noteworthy that the final position in the deliberative mode is not guaranteed to be as specified for reasons of robustness. Hence it is important for the deliberative planner to incorporate uncertainty in final leg and body pose. Such planning will generate more fault tolerant steps.

10.4.2.1 Walking Scheme

Objective: To move as close to the goal as possible using (robust) reactive behavior.

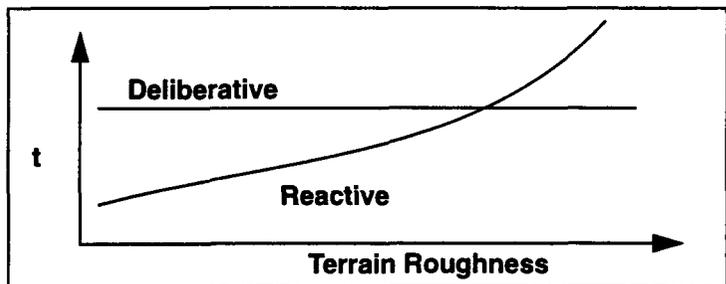
- Guided by cost function $f(\text{energy}, \text{error})$.

$$Energy = f_1(t, Power); Power = Max \Rightarrow Cost = G(t, error) + PoseCost$$

$$G = \sqrt{\left(\frac{t}{t_n}\right)^2 + \left(\frac{e}{R}\right)^2}; t_n = \text{nominal}; \frac{e}{R} = \left(\frac{\text{error}}{\text{Step}}\right)$$

- Cluttered field is defined by crossover time between walking and deliberative walking

$$Te = t_{OLD} \times 0.9 + t \times 0.1$$



- Use 2.5 D uncertainty grid. Each cell consists of z value and certainty/evidence. Grid size = 1/8 of foot diameter

- Pick the feet up to the 1.2 x highest obstacle in stereo map and move body
- Constraint : Keep feet 20 cm away from known obstacles
- Constraint : Ensure stability margins by periodic checks
- Constraint : Correct body pose to within 2 deg every step
- Utilize behavior to get near optimal time performance. This statement assumes that good performance can be obtained by using (clever) behavior rules.

10.4.2.2 Effect of Walker on Search Strategy

Priorities to enable global/local planning and search to integrate walking behavior into path optimization:

- Leg motions < Body rotations < Body acceleration < Vertical motions < Slope traverse < Robot eating canyon. These qualitative remarks can be quantified by power considerations.
- Classify terrain in terms of traversibility and hazard. Use these measures in A-Star search.
- Traversibility includes obstacles, slopes, forgiveness to errors.

EX1 :A smooth path 3m wide with drops on either side is BAD. However an identical path with steep sides is very good.

EX2: A 20m detour is preferable to a 10m path with a 1m climb and descent.

10.4.2.3 Stereo Strategy

Objective: Streamline stereo processing to maximize resolutions in the given time. Utilize ability to change resolutions on the fly.

- Resolution suggested = half foot size x 10 cm (or a practicable).
- Take snapshot at the end of body motion as legs go down. Least vibrations, pose almost ideal.
- Take snapshot during middle of walking cycle before deceleration. More processing time, non-ideal pose, vibrations (long. and lateral). Design mast for vibrations? How do vibrations affect stereo?
- Pan and tilt may be required if second option is used.

The field of view and camera orientation requirements for the local planner and reactive walking are vastly different. In addition, it is computationally infeasible to generate 5cmx5cmx5cm local maps in the time taken for a single step. Several other stereo strategies were similarly discarded. The current status for stereo usage is to use it to generate preliminary evidence maps. Using a pan-tilt mechanism, it is possible to generate fine terrain maps for the current step. This time needed is of the order of a minute.(Quote figures?)

10.4.2.4 Planning Information Requested

Objective: Use planning information to make educated guesses to intent of motion.

- Direction of motion, step size, final pose
- Detailed final pose if in deliberative mode
- Constraints like max acceleration on slopes

Other information listed below is not necessary, but can increase walking efficiency:

- Intended local direction (pure pursuit direction)
- Information on niceness of neighborhood (5m)
- Constraints on viable directions if applicable

10.4.2.5 Control

Objective: Incorporate behavior into control loops to reduce complexity and increase response.

- Fast PID not needed due to slow overall motion and low trajectory accuracy requirements
- Fuzzy mapping gives ability to integrate kinematic, dynamic and actuator models without overhead.
- Fuzzy logic controllers can seamlessly integrate exception handling into the control loop and provide fast response without using interrupts.

To generate fuzzy maps, a simplified but complete dynamic model is required. Such a model can be obtained by developing a complete model and deleting higher order velocity terms. Using Kanes dynamics (as used by Adams) to generate the model will yield a computationally efficient model. If a fuzzy controller is not used, using model feedback for decoupling the control directions can be carried out.

Safety interrupts are needed either as fuzzy rules or as exceptions. These exceptions will be classified into ones that require stopping and notifying the superior module, ones that require notification but where waiting for response is unnecessary and ones that can be entirely handled by the reactive module. This will help increase walking efficiency.

10.4.2.6 Sensors

If the conclusions drawn regarding stereo are believed, augmenting stereo range & uncertainty images with fast, local range sensors is a must. The various sensors considered for walking in various modes are:

- Contact sensing of collision by bumper device around the foot perimeter
- Foot contact sensing by compliant pad with embedded sensors or by piezoelectric pads
- Proximity sensing using a capaciflector array arranged in a hemispherical orientation for both peripheral and downward directions.
- Light striping sensors of the kind developed by the MEASUR project.
- Lookdown VLSI range image sensor developed by Andy Gruise and Takeo Kanade.
- Use encoder readouts combined with actuator voltage and power readings to estimate forces and determine contact.

Other, non-conventional sensors:

- Whiskers for proximity sensing.
- Pulsed laser-intensity sensors on each leg for range sensing.
- Extra reconn sensors on beam to accommodate body.

Array and area sensors were primarily considered since point range information is insufficient and not robust.

10.4.2.7 Computing & power

The target is to use 3/4th computing power of 68020 for reactive planning. Additional resources will be used for motion control. The goal for sensor power is < 4 W.

10.4.2.8 Other Suggestions

Reactive Body balance can be combined with foot placement, since significant body drift is unlikely in a single step. Impact sounding mode to detect and avoid hollow terrain patches can be added for lava tube exploration.

10.5 Simulation

In the final stages of system testing, there is undoubtedly no substitute for using the actual Daedalus robot. However, the existence of a simulator for the robot will aid development of perception and planning software in several ways: it will make debugging easier, since hardware errors can be ruled out in simulation; it will allow the software to be tested before the robot is fully constructed; and it will allow for cheaper testing of software than a real-world robot run.

Three basic units will form the simulator. The first is a simulation of the environment in the form of a topological map, possibly augmented with some simple information about terrain type. The second is the Mechanism Simulation unit; this is a simulation of the actual robot mechanism, which will attempt to duplicate the perception and responses of the true robot were it to be run on the simulated terrain. The third is a collection of dummy software modules, which will provide a way for actual software modules to be tested individually. A user interface will allow some of the parameters of these three units to be altered dynamically, and will provide a way to display the state of the simulated robot in real time.

The simulator is run by connecting the Control module to the Mechanism Simulation Unit instead of to the Daedalus. Optionally, some of the other modules may be replaced by dummy modules; conditions under which these dummy modules would be substituted are described in section 9.2.3.

The most important characteristic of a simulator is that its' existence be transparent to the software being tested. Thus, the interface to the Mechanism Simulation Unit must be precisely the same as the hardware interface to the Daedalus, and the interfaces to the dummy software modules must be the same as their real counterparts. This transparency prevents programmers from having to worry about their software functioning in two different environments (simulated and real); and it also provides some assurance that a module which functions correctly in simulation will perform similarly on the actual mission.

A secondary, but still important, characteristic is the simulator's running time. While it is not necessary for the simulator to run in real time, it should be practical for the robot to move several hundred meters in simulation; path lengths of this magnitude are necessary to test the capabilities of the global and local planners. It is therefore important to keep the computational complexity of each simulator operation as low as possible. Of course, simulator time steps will be discrete, and the running time can be improved by increasing the length of a discretized time unit; but this will result in a less accurate simulation.

A further consideration is accuracy. What counts as sufficient accuracy depends on what the simulator is intended to test; when testing the path planner, for instance, a very rough approximation may suffice; much higher fidelity would be required if the controller were to be meaningfully tested. There is a tradeoff between running time and accuracy, and it will probably be best to let the user decide the necessary accuracy for a particular simulation.

Even in its slowest mode, there are many phenomena the simulator will ignore. For instance, no attempt has been made to model vehicle dynamics, friction forces between the robot and the terrain, deformation of robot components under stress or temperature, or numerous other factors which will plague the real robot. Modeling these affects is deemed to be difficult, unreliable, computationally expensive, and of limited use to the software debugging task.

10.5.1 Environment Simulation

One possibility for the environment simulation is to use a topological map of an actual location. However, the necessary resolution for this unit is probably finer than 1 square meter, and there are probably few (if any) locations which have been mapped out to this precision.

There are currently techniques in existence for generating detailed, realistic artificial topological maps using fractal models of natural terrain. These techniques could be used either to generate an entirely fictitious terrain, or to fill in an actual map to the necessary level of detail. The use of these techniques has the advantage that simulation can be performed on terrain of arbitrary difficulty simply by tuning the parameters of the landscape generator.

One simple method for generating fractal terrain is as follows:

- 1) At fairly sparse grid points, height values are assigned by adding noise of a given magnitude n to a constant.
- 2) Midpoints of the grid are assigned height values by linear interpolation of their neighbors to form a grid of twice the resolution.
- 3) n is set to $n/2$, and noise of this new magnitude is added to all grid points.
- 4) If the grid is of sufficiently high resolution, stop; else go to step 2.

Figure 9-2.1 shows a cross-section of the terrain generated by this algorithm over the first four iterations.

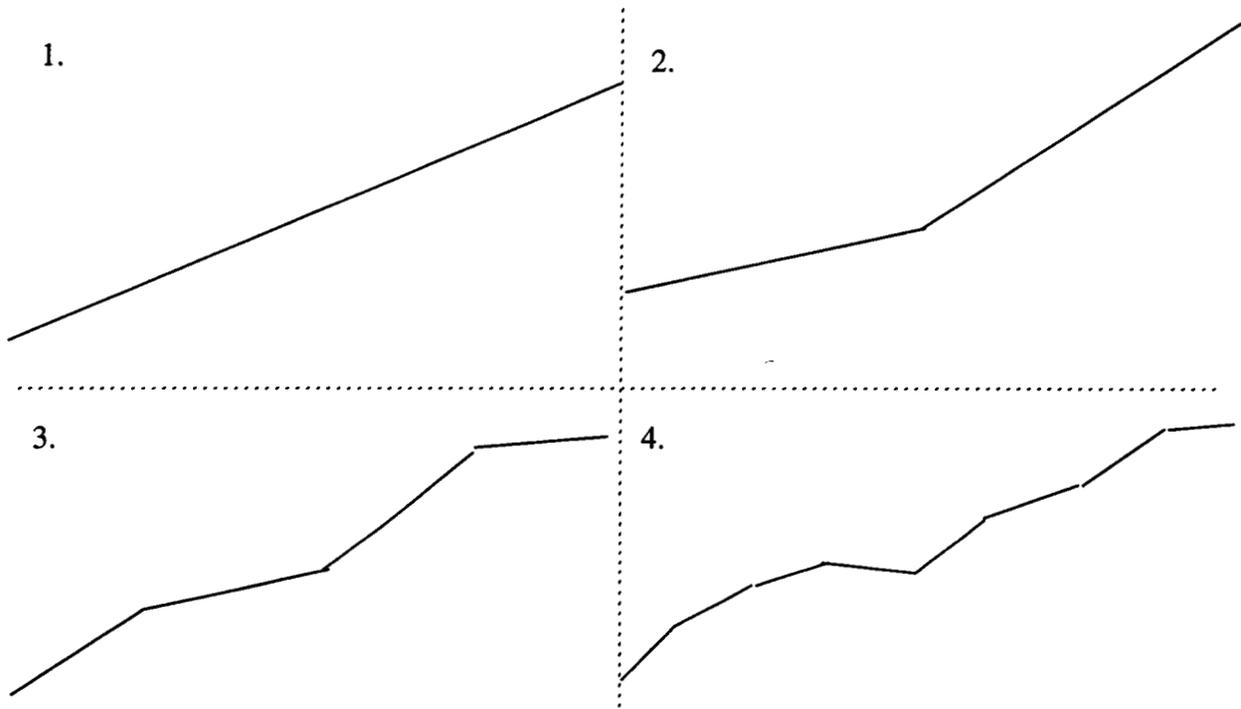


Figure 10.5-1An example of fractal terrain generation

In addition to a topological map, it may be useful to simulate terrain type at a very rudimentary level. For instance, a “hardness” measure of the terrain could be used to determine to what depth the Daedalus robot’s legs would sink while walking.

10.5.2 Mechanism Simulation

There are two tasks included here: updating the pose of the robot on the simulated terrain after actuator commands, and generating simulated sensor output.

10.5.2.1 Pose Simulation

At each discrete time step, the Mechanism Simulation Unit must:

- simulate the incremental effect of actuator commands on the robot,
- check for collisions with terrain, and
- verify the stability of the new pose (i.e., checking for tip-over).

Since we can safely ignore vehicle dynamics in a robot moving only a few meters per minute, only the kinematics of the Daedalus must be considered in the first step..

Checking for collisions with terrain could be an expensive process, since the robot has curved surfaces and the terrain is complex. It would be simplified immensely by assuming that any possible collisions will occur with the robot's feet; it is possible to design scenarios where this is not the case, but they are few and far between, and generally only occur in situations which the Daedalus should avoid. Collision checking is probably best done by locally fitting polygons to the terrain and to the Daedalus' feet, and checking for intersections. An advantage of this method is that the number of fitting polygons used can be adjusted up or down, depending on the need for accuracy vs. running time.

Verifying the stability is a simple matter of checking if the center of gravity of the robot is over the support polygon formed by the legs; this is an inexpensive process.

10.5.2.2 Sensor Simulation

Output from robot state sensors such as inclinometers or gyroscopes can be simulated directly. This is also true of environment sensors such as sonars or laser rangefinders.

Accurate simulation of camera output is extremely computation-intensive and probably not worth the cost. However, it is possible to simulate the effective information gain provided by a camera; for instance, by simulating the output of the image processing module rather than the output of the actual camera. This solution has the disadvantage that it does not actually test the image processing module, since this module would be cut out of the loop entirely. Since a simulated robot run would probably not be a particularly effective method for testing an image processing module in any case, this drawback is not significant.

Nonetheless, there may be cases when it is desirable to test the image processing module within the simulator (perhaps to test its interfacing with other modules). Using the true image processing module on accurately rendered images will likely be too slow. However, a hybrid perception module, which uses the simulation method described above most of the time and only occasionally calls the true image processing code on rendered images, would provide some testing ability at a more feasible computational cost.

10.5.3 Software Module Simulation

The first two simulator units would already provide significant capability for testing software. Without some additional components, it would be necessary to test all of the Daedalus's software simultaneously; however it would be useful to run software modules in isolation as well.

One way to accomplish this is to create a dummy for each of the software modules. If some set of software modules needed to be tested independently of others, only the modules of interest would be real; the others would be replaced by the corresponding dummy modules. These dummy modules must be simple enough to be reliable, and they must provide output which could plausibly have been generated by an error-free version of the corresponding actual module.

Practically, some modules do not lend themselves well to replacement by dummy modules. It would be difficult to design a dummy Base Station Manager, for instance, or a dummy Data Manager; and dummy modules of this type are unlikely to be useful for debugging in any case. How-

ever, dummy modules for the Path Planner, the Trajectory Planner, the Image Sensing Manager, and the Controller are both feasible and likely to be useful.

An example configuration of the simulator is shown in figure 10.5-2 (only pertinent modules are shown). This configuration will test the Path Planner in isolation from the ISM ,the Trajectory Planner, and the Controller.

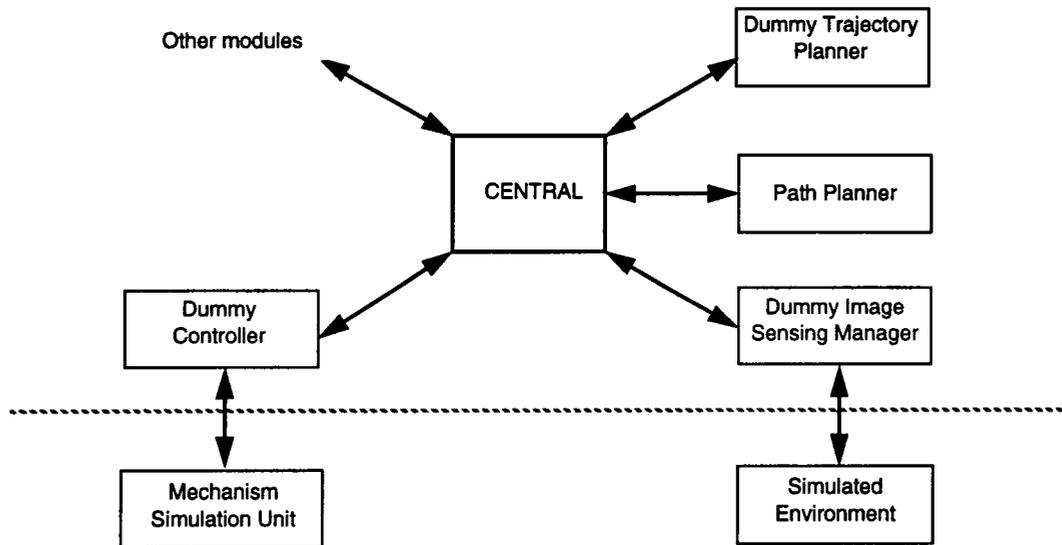


Figure 10.5-2 Sample Simulator Configuration

10.5.3.1 The Dummy Image Sensing Manager

The actual ISM will take input from the sensors and update the internal representation of the environment to be used by the planning modules.

The dummy module would provide the same output, but it would take input directly from the Environment Simulation unit, rather than from simulated sensor readings. Since the dummy unit would have access to the actual simulated environment, it would be capable of providing perfect output.

10.5.3.2 The Dummy Path Planner

The actual path planner will get input from the trajectory planner, sensor readings, and the internal representation of the environment. It will then do one of the following:

- Generate a series of foot movements in the direction desired by the high level planner and send commands to the robot, or
- Generate a detour around some local obstacle and then generate foot movements along this new path, or
- Fail, and inform the trajectory planner that the path is blocked.

Any resulting foot movement commands will be sent to the robot controller, or (if in simulation) to the Mechanism Simulation unit.

In most instances the dummy module would update the simulated robot's pose directly, bypassing the Mechanism Simulation unit. Occasionally it would randomly either plan detours and then update the pose, or fail and notify the high-level planner. Thus the dummy module would test the ability of the trajectory planner to deal with failures and detours.

10.5.3.3 The Dummy Trajectory Planner

The actual trajectory planner will get input from the internal representation of the environment and from the path planner. It will then output a desired route to the path planner.

The dummy trajectory planner would simply output random desired paths; this would be sufficient to test the path planner.

10.5.3.4 The Dummy Controller

The actual controller is primarily responsible for three functions: body balancing, gait generation, and reactive foot placement.

The dummy controller will have to do body balancing, but since it has access to the robot's actual pose unfiltered by sensors, it can accomplish this merely by setting the robot's pose correctly. The same is true of gait generation. It seems unlikely that reactive foot placement can be performed simply and efficiently by a dummy module, so the actual reactive foot placement software would have to be included in the dummy.

Results from simulations using the real controller will only be meaningful if the accuracy of the simulation is fairly high, in which case the simulator will run slowly. The dummy module should probably be used in all cases in which the controller itself is not being tested.

10.5.4 User Interface

The user interface consists of a set of displays of the state of the simulated robot, and a set of operations which can be performed by the user at run-time to dynamically alter the robot's state.

10.5.4.1 Displays

While running, the simulator should be capable of displaying the following information:

- A rendering of the robot in its current configuration on the terrain,
- The local elevation map as constructed by the ISM, and
- The current path being followed, showing the actual robot location as well as the robot's internal estimate of its position.

An example of how the user interface would appear is included in figure 10.5-3.

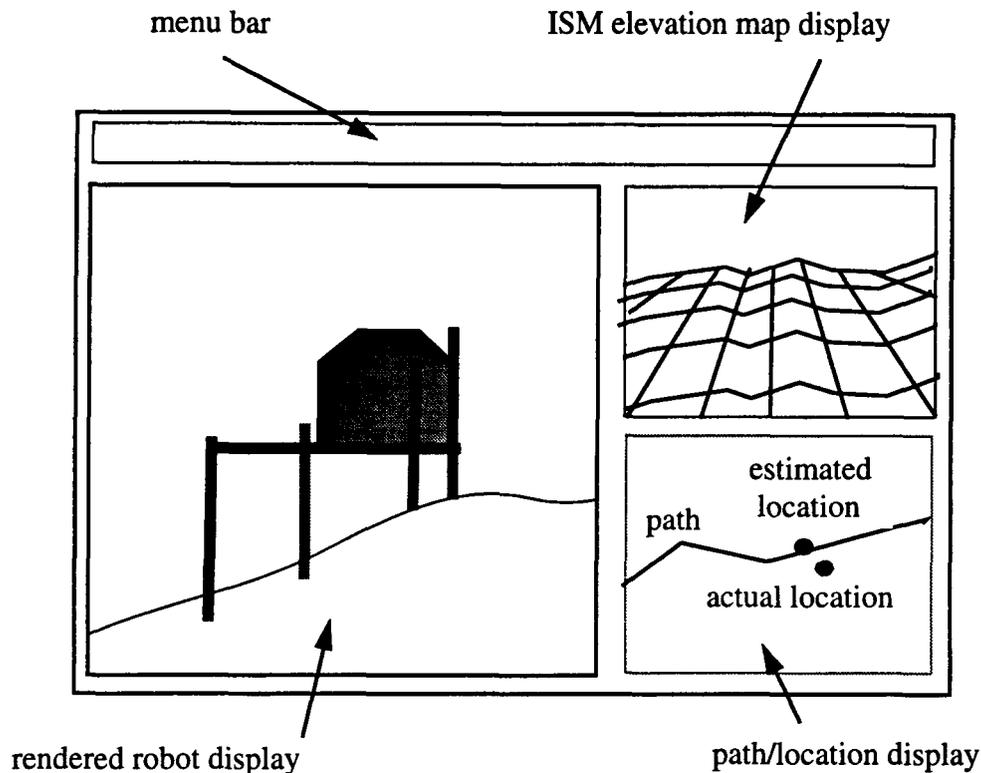


Figure 10.5-3 Graphical User Interface

In order to prevent the simulator from spending an excessive amount of time updating the displays, the update rate of each display should be controllable by the user at run-time. Specifically, the user should have the option to update a display continuously or periodically either in time or in distance traveled by the robot. It would also be useful to be able to program these display rates so that the simulator could carry out commands like “update the rendered robot display every 50 steps for 100 meters, and update it continuously thereafter.”

10.5.4.2 Run-time Capabilities

There are a number of capabilities which would be useful, and require minimal programming effort. These include:

- Changing the length of a discrete time step: the user should be able to control this parameter directly, and also to program it to vary with the gait stage; i.e., it may be desirable to use a large time step while picking up the leg, and a small one while setting it down.
- Decreasing/increasing the accuracy of the simulation in order to increase/decrease the running time: this parameter would control the number of polygons used to fit the robot and the terrain in the collision checking stage..
- Altering the robot’s nominal position: the user should be able to directly alter the robot’s estimate of its position. This would be useful to test the ability of the Daedalus to recover from position errors.
- Altering the path the robot attempts to follow, and adding or deleting obstacles to the simulated terrain: the user may wish to lay out a “test course” to test the Daedalus’ ability to navigate specific obstacles.

- Replaying a specific portion of a simulation. This capability requires that at each point in time, the path being followed and the nominal & true position of the simulated robot be recorded.
- Simulating failure of a component (such as the cameras) for a period of time. Other components whose failure can be easily simulated include inclinometers, leg actuators, and the global positioning system. Component failures could be simulated by addition of large amplitude Gaussian noise to the component behavior, or by return of completely random readings.

10.5.5 Conclusion

This simulator design is unlikely to meet anyone's conception of the "ideal" simulator. There are certainly many features which could be added or embellished. The degree of realism in the system and the quality of the rendered graphics could be tinkered with eternally. However, the implementation and run-time costs of new features should be carefully weighed against the benefits they provide; after all, time spent improving the simulator is time taken away from other, perhaps more critical, tasks. A simulator is a tool intended to make the debugging task easier; it is not a final product. This design is thoroughly grounded in the assumption that the existence of the simulator is, in the end, justified solely by its contribution to a successful Daedalus mission.