

CraterGrader: Autonomous Robotic Terrain Manipulation for Lunar Site Preparation and Earthmoving

Ryan Lee^{*†}, Benjamin Younes^{*†}, Alexander Pletta^{*†}, John Harrington^{*†}, Russell Q. Wong^{*†}, William “Red” Whittaker[†]

^{*}Equal Contribution, Listed in Random Order

[†]Carnegie Mellon University.

Abstract—Establishing lunar infrastructure is paramount to long-term habitation on the Moon. To meet the demand for future lunar infrastructure development, we present CraterGrader, a novel system for autonomous robotic earthmoving tasks within lunar constraints. In contrast to the current approaches to construction autonomy, CraterGrader uses online perception for dynamic mapping of deformable terrain, devises an energy-efficient material movement plan using an optimization-based transport planner, precisely localizes without GPS, and uses integrated drive and tool control to manipulate regolith with unknown and non-constant geotechnical parameters. We demonstrate CraterGrader’s ability to achieve unprecedented performance in autonomous smoothing and grading within a lunar-like environment, showing that this framework is capable, robust, and a benchmark for future planetary site preparation robotics.

I. INTRODUCTION

As humanity sets its sights on the Moon, Mars, and beyond, human presence requires surface infrastructure similar to the buildings and roads relied upon for millennia on Earth [1] [2]. Robotic systems show promise in constructing surface infrastructure in aerospace applications, including landing pads, roads, structural foundations, trenches, and berms, particularly in locations such as the Moon or Mars. However, the technological demands of operating in the low-mass [3], low-energy, and high-stakes environment of space require new approaches to robotic construction. Challenges such as time delays and lack of environmental awareness in teleoperation, can be overcome through increased autonomy [4]. Continuous operation time is crucial when vehicle lifetime and operationally-constrained energetics are limited. To this end, autonomy is capable of operating patiently and continuously over a 14-day lunar daylight period with minimal human supervision. Thus, there is a pressing need to develop autonomous robots that can efficiently and effectively perform lunar site preparation.

We present CraterGrader: a novel approach to autonomous robotics for lunar site preparation (Figure 1). The main contribution of this work is a combination of the following: (1) System demonstration of autonomous lunar grading meeting NASA’s LuSTR RFP specifications [2] with no priors, (2) Novel application of optimal material movement planning for autonomous construction, and (3) GPS-free precision robot



Fig. 1. CraterGrader in the MoonYard. Top left: Pseudo-sun fiducial tag used solely for bearing estimation. Center: CraterGrader. Top right: Leica Viva TS16 Total Station.

localization subject to planetary site preparation constraints. We also discuss the experimental performance and results of the manifested robotic system, emphasizing its potential applications in lunar infrastructure development. Our work demonstrates an approach to delivering a site preparation robot that can autonomously manipulate lunar-like regolith in a worksite to form desired topography.

II. RELATED WORK

A. Earth Moving Autonomy

Site preparation refers to the tasks that produce a baseline worksite for building foundations and superstructures. Flattening and smoothing the terrain to meet a specified planar angle, also called “grading”, is a time-intensive component of construction that requires precision. Although grading is a common task in most construction projects, there is limited research on autonomy in this area, despite significant industry interest [5].

Early work demonstrated an initial framework for intelligent earth working [6], focusing on developing a system architecture and hierarchy of the proposed concept. Bonchis

et al. [7] developed an autonomous skid-steer loader, supported with a sensor suite, onboard computer, and limited supervised autonomy, highlighting the potential of an automated solution in the task of bulk material handling. Later, Marshall et al. [8] and Schmidt et al. [9] presented a number of approaches to the dig and dump planning processes, using a 2.5D (height-field) representation of the environment to plan digging. Jud et al. [10] developed a robotic walking excavator, HEAP, retrofitted from a commercial excavator and demonstrated autonomy in excavation tasks, including the formation of free-form embankments. State-of-the-art systems deployed in the construction industry have focused primarily on assistive features [11] or selected subtasks within the greater construction industry [12].

Unlike terrestrial priors, our approach is energy-efficient within the constraints of lunar applications and does not rely on existing infrastructure such as GPS.

B. Lunar Construction

The field of lunar construction and In Situ Resource Utilization (ISRU) has grown significantly in the 21st century, as the goal of a presence on the Moon has become more in reach [1]. In 2004, Muff et al. [13] demonstrated an initial prototype bucket wheel excavator. This was later expanded upon by the NASA Swamp Works team at Kennedy Space Center in 2010 [14], where the RASSOR platform was developed using the principle of a counter-rotating bucket drum to enable strong cutting forces with reduced weight-on-tool. Lunar site preparation was explored in 2009 during the integration of a front-facing lightweight bulldozer blade onto NASA's Chariot platform [15].

Recently, more attention has been given to developing lightweight robotic systems [16]. In 2014, a lightweight robotic excavator was demonstrated to carry over 50% of its weight [17]. In 2018, a teleoperated platform was developed for site preparation that graded and leveled a basalt rock simulated lunar worksite [18]. A rare example of autonomy research within lunar excavation focused on an evolutionary learning approach to planning, as well as ground truth localization and mapping information provided from an external source [19].

In 2021, a NASA Lunar Surface Technology Research (LuSTR) request for proposal (RFP) outlined a desire for an autonomous lightweight system that could grade, compact, and remove rocks from a lunar worksite for a lunar landing pad [2]. The RFP emphasized the lack of research in lunar site preparation autonomy and provided a standard reference for lunar site preparation requirements. These requirements directly inspired this project's goals and system design, providing a framework for effective evaluation.

C. Lunar Localization & Mapping

Localization is one of the most difficult challenges within planetary robotics due to the lack of infrastructure, harsh lighting conditions, and unknown environments. Past missions,

such as the Mars Exploration Rovers, heavily relied on Visual Odometry [20] and a higher tolerance for error. Ultra-wideband (UWB) beacons are of increasing interest for Time Distance of Arrival positioning in lunar environments [21].

Other localization methods include using neural networks for global map correspondence [22] and crater landmark detection [23]. However, in the context of site preparation, relying on such landmarks for localization is not feasible as they would be inherently altered during terrain manipulation processes. Molina et al. [24] investigated the use of robotic total stations for line-of-sight tracking as part of a lander subsystem. A robotic total station was also used in a lunar prospector platform, Scarab [25], resulting in precise localization and robust slip estimation at 1 Hz. Strict vertical height tolerances for mapping and grading necessitate 3D positioning to a precision that is unprecedented in prior planetary robotics applications.

D. Optimal Transport

Optimal transport refers to the problem of transforming one distribution into another using an *Optimal Transport Plan*, which minimizes the cost of transforming between the two distributions, also known as the Wasserstein distance. Gaspard Monge conceptualized the field of optimal transport in 1781 for transportation theory in the context of soil transport [26]. Optimal transport has practical applications in fields such as networking [27], image retrieval [28], graphics [29], and computer vision [30] [31]. An optimal transport plan represents a function map that transforms between the distributions through a minimally exertive process, analogous to mechanical work. Zhai et al. [32] used optimal transport maps for path planning in unknown environments. The name “Earth Mover’s Distance” (EMD) was coined by Jorge Stolfi in 1994 and is synonymous with the Wasserstein distance [28]. The term “Earth Mover’s Distance” closely matches our work in a most literal sense, in which we introduce a novel application of EMD in the field of robotics and earth moving.

III. SYSTEM ARCHITECTURE

A. System Objective

The main objective of CraterGrader is to change the local grade, absolute height, and surface smoothness for a given *worksite* of previously unseen lunar-like topography into the desired flat terrain topography. Here, the term *grade* refers to the slope, or gradient, of an assumed continuous surface topography. Grade, measured in degrees, is defined as the angle between a fit plane to the terrain and a level plane normal to gravity. Smoothness, measured in cm, is computed as the standard deviation of the distribution of positive and negative terrain heights relative to the fit plane. Visualizations of these metrics against a crater cross-section are shown in Figure 2. To measure the performance of the system, we derive post-operation grade, with respect to the ground plane, and smoothness requirements of $\pm 1^\circ$ and 1 cm respectively directly from the LuSTR RFP as a baseline [2]. We define an additional metric, area out-of-spec (OOS) that measures

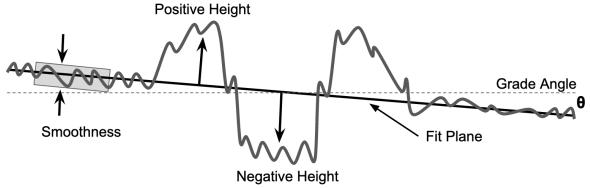


Fig. 2. An example crater cross-section, depicting key parameters such as grade, smoothness, and positive/negative height, with respect to a fit ground plane. In particular, grade and smoothness metrics are used to evaluate CraterGrader’s performance against the LuSTR RFP baseline [2]. Grade is defined as the angle between the fit plane and a plane normal to gravity. Smoothness is calculated by constructing a distribution of positive and negative terrain heights relative to the fit plane and computing its standard deviation. The roughness of the topography and grade angle are not to scale and are exaggerated for effect.

the worksite area outside of the aforementioned grade and smoothness requirements, reported as a percent reduction after autonomous operation.

The *worksite* is defined as a rectangular area where work is to be completed. To simplify practical operations, we assume that the worksite has obstacle-free, homogeneous terrain and no other agents or gravity offloading. The parameterization of craters in the worksite assumes a depth-to-diameter ratio of 0.2 m and a crater diameter of 1.0 m, derived from the LuSTR RFP [2], and additionally assumes that the volume of material removed from the crater is equal to the volume contained in the crater rim.

The concept of operations is as follows: CraterGrader is placed in the worksite and is given a high-level command to begin autonomous operation. This comprises an initial exploration phase, where no mapping priors are used, followed by a transport phase during which bulk material is moved while grading and smoothing, described in detail in Section IV. The robot’s autonomous operation ceases once all the transport objectives are accomplished. Additionally, the system generates a human-interpretable topographical map of the current worksite, which can be utilized for autonomous planning and also update remote human operators on grading performance and progress.

B. Robot Platform

The robot mobility platform, illustrated in Figure 1, has a 0.5 m wheelbase and 0.43 m center-line track width, which fall within a sub-25 kg total mass. The chassis is made of aluminum extrusion and serves as the backbone for the roll-averaging rocker suspension. The wheels consist of a polymer exterior and foam core, but can be swapped with metallic wheels for flight applications. The wheels are driven through the front and rear differentials, with independent steering of the front and rear axles in a double Ackermann configuration. On flat ground, the robot produced 138 N of drawbar pull, as measured by a force transducer. The robot is powered by a rear-mounted battery system, featuring a 20 V 15.0 Ah Li-ion battery with regulation and fusing, which is sufficient for the

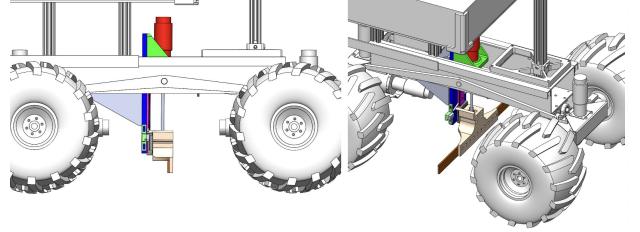


Fig. 3. The CraterGrader belly-mounted grading blade assembly. The actuator shown in red is kept inside the robot chassis perimeter to be protected from dust and rock strikes. The blade translates vertically and is rigidly constrained to a linear rail bearing.

100 Watt maximum load on the system. At the front of the robot, a vertical mast provides mounting for various perceptual sensors, and its highest point is approximately 0.7 m from the ground.

The remaining electronics are housed in a dust-protected compartment on the top of the chassis. This compartment includes an NVIDIA Jetson AGX Xavier Development Kit as the main compute system for running the autonomy stack and integrating all peripheral sensing components. An Arduino Due board serves as a lower-level interface for the motor controllers and communicates with the Xavier over a serial connection. All compute systems utilize ROS 2 Galactic middleware to facilitate inter-process communication, with micro-ROS bridging the Xavier and Arduino Due boards.

C. Grading Blade

The bulk of CraterGrader’s material transport is performed by the grading blade, shown in Figure 3. The grading blade has dimensions of 250 mm in length, 50 mm in height, and 3 mm in thickness. The flat blade takes inspiration from terrestrial motor graders and is center-mounted for advantageous drawbar pull, a precious resource in diminished gravity as well as reduced tool gouging on pitched topography. The blade is actuated vertically and can make cuts in the terrain at specified design heights for surface grading and filling craters. This vertical degree of freedom with fast response time enables the grading blade to react dynamically to the terrain as a function of the measured slip or desired cut, and furthermore provides a fallback mechanism for escaping high-centered scenarios. Furthermore, the yaw and pitch angles of the blade can be adjusted statically to experiment with further configurations. In practice, however, maintaining a default blade orientation parallel to the vertical plane yielded the best results, enabling a configuration close to under-vehicle dozing. The grading blade height is relatively shallow to reduce the likelihood of slippage and embedding, allowing excess material to flow over the blade, to be transported in later cuts.

D. Drag Mat

CraterGrader features a rear-mounted chain mail drag mat, as seen in Figure 1. This drag mat effectively smooths high-frequency surface variations, utilizing a similar technique employed in sporting events to smooth terrain, such as a

baseball park infield. Notably, the 316L stainless steel drag mat is essential for smoothing out impressions and tire tracks left by the robot during its traversal of the worksite. Although not applicable to all site preparation tasks, our experiments have shown that this drag mat is effective in smoothing out the terrain. Moreover, it is fully passive, simple, and flight-facing with the use of stainless steel.

E. External Infrastructure

Testing for lunar site preparation was conducted in a 56 m^2 sandbox referred to as the *MoonYard* (Figure 1). The MoonYard features a local network for communication between the robot, the robotic total station, and operators. The regolith simulant used during sandbox testing was fine-grained washed quartz sand with an approximate angle of repose of 31° and bulk density of $1603 \frac{\text{kg}}{\text{m}^3}$. These capabilities facilitate thorough testing and analysis of lunar excavation techniques in a simulated lunar environment.

To provide externally referenced localization estimates, the worksite utilizes a robotic total station and fiducial tags that surround the site, which act as targets for a pseudo-sun tracker. Their usage is described further in Section IV-A. An NVIDIA TX2 compute system interfaces between the robotic total station and the robot in the worksite, communicating through the local area network.

IV. AUTONOMY

A. Sensing

CraterGrader's sensing suite is designed to operate within the constraints of a lunar environment. An externally-positioned robotic total station tracks a retroreflective prism mounted to the sensing mast of the robot to provide an accurate absolute 3D position estimate with millimeter precision. We present a novel application of the robotic total station to lunar construction, inspired by research application in terrestrial site preparation [33]. This approach is motivated by the absence of GPS and a desire for site-wide consistency. The total station used is a Leica Viva TS16 robotic total station, and is modified to stream measurements at a rate of 7-10 Hz. A flight-forward approach could include a deployable external sensor payload or an additional mobile base for total station setup. A VectorNav VN-100 AHRS (Attitude and Heading Reference System) measures linear acceleration and angular velocity to generate roll and pitch angle estimates. To obtain absolute bearing measurements, sun-tracking sensors are typically used in space applications [34]. CraterGrader utilizes a pseudo-sun sensor to mimic this modality in an indoor testing environment by means of a fiducial-tracking fisheye lens camera. An Intel RealSense D435i camera serves as the primary perception sensor, leveraging passive stereo reconstruction to generate point cloud data for terrain analysis. Active stereo through IR illumination was avoided due to infrared scattering expected while operating in dusty lunar terrain. Although lunar dust could degrade depth data in a passive stereo camera, the impact observed during testing was minimal and did not require further mitigation measures. While not space-rated, the

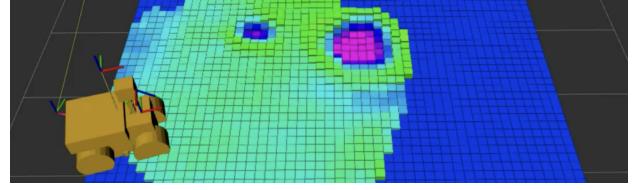


Fig. 4. The worksite map is updated as new perception data are acquired. The grid cell color is overloaded in this figure to show both the height of the mapped grid cells and whether a cell has been observed. The dark blue shows map grid cells that have not yet been observed by the robot. Two observed craters are shown alongside the 3D robot model.

RealSense reflects a feasible sensor modality for lunar perception as stereo vision operating in the visual light spectrum. In this mode, the RealSense requires external illumination, and our testing assumes a bright, overhead-lit environment. Lunar deployment may necessitate additional considerations, such as external site lighting or self-illumination on the rover. Explicit crater detection is not applied, allowing for a generic representation of the lunar terrain.

B. Localization

Operating in a lunar-like environment presents unique challenges for localization in that: (1) the environments are near-featureless, and (2) any prevalent features such as craters are to be destroyed during operation (with the end goal being a featureless, planar surface). Consequently, robust localization that does not rely on traditional SLAM or feature-based methods is required. The 3D pose is estimated by fusing *feature-agnostic* sensor measurements, including live robotic total station positioning, AHRS, pseudo-sun tracker, and on-board encoder telemetry data. Two Extended Kalman Filters (EKFs) are run in parallel to compute both an accurate global position in the *map* frame as well as a smooth, continuous velocity in the *odometry* frame, implemented via the ROS 2 `robot_localization` package [35]. Robot slip is estimated by comparing odometry-measured velocity relative to the absolute position as measured by the total station.

C. Perception & Mapping

Point cloud information from stereo reconstruction and state estimates from localization are utilized to generate a 2.5D discretized map of the worksite terrain through a filter-based approach. The system transforms reconstructed 3D points into the map frame and bins them into map grid cells, where each map cell is modeled as a one-dimensional Kalman Filter, akin to Cremean and Murray [36]. This binned 2.5D map structure provides a lightweight representation of terrain topography which, compared to higher-density point cloud registration methods, is more robust to operating in an environment with sparse features. The confidence estimate of newly seen points is based on localization confidence and measurement distance from the stereo pair. After a cell has been observed a number of times, the robot is increasingly certain about the height in the grid cell until the agent disturbs it with its blade or tires. We model this as noise injection into the grid cell filters after

the robot has driven over or manipulated the terrain of those cells. The resulting 2.5D grid map, seen in Figure 4, represents the expected heights at grid cell locations as seen by the robot.

D. Planning

The planner generates high-level task plans and calculates trajectories for tool operation and the mobility platform to manipulate the terrain based on the robot's state and environment. At the core of the planner is the Behavior Executive, which manages system actions by interfacing with various planners and other software modules. The Behavior Executive also monitors the trajectory-following process and system health, as well as handles faults. The executive is implemented as a hierarchical finite state machine with two levels to switch between operating mode and individual action states, providing a structured approach to the planning and execution of the system's tasks.

The high-level planning strategy involves an initial traversal of the worksite to build up a map of terrain topography prior to earthmoving, followed by an optimal transport plan using the terrain map to perform earthmoving. These steps are handled by the Exploration Planner and Transport Planner respectively, as detailed below.

1) Exploration Planner: Prior to grading, a full map of initial terrain topography is necessary to perform optimal material transport planning as described in Section IV-D2. This is achieved by the Exploration Planner, which generates the plan to navigate a worksite of unknown terrain and ultimately build a terrain topography map. First, an operator provides the Exploration Planner with knowledge of the rectangular worksite, including dimensions (length and width) and location of the worksite relative to the robotic total station. The Exploration Planner then generates a set of geometrically-defined waypoints to first navigate the worksite boundaries and then cover the center, to ultimately achieve full map coverage. This strategy proved to be more feasible for the robot's kinematics and large turning radius, compared to other approaches such as a naive raster pattern.

2) Transport Planner: The Transport Planner calculates which *highs* of the worksite should be pushed to which *lows* to convert an arbitrary starting worksite topography to an arbitrary final design topography. Full details on this problem formulation are provided in Section V.

This approach was selected instead of motion parameterized by crater geometry to robustly handle malformed craters (as created during operation). Additionally, this approach allows for re-planning for self-corrections, and seamless extension to generalized terraforming. Within the context of grading craters, a design topography was created as a flat plane so the result was a *transport plan* II to pair waypoints along crater rims (*highs*, referred to as *sources* as they contain more material than desired) to waypoints along crater floors (*lows*, referred to as *sinks* as they contain less material than desired), with corresponding material transport volumes. An example transport plan for a real single crater is depicted in Figure 6.

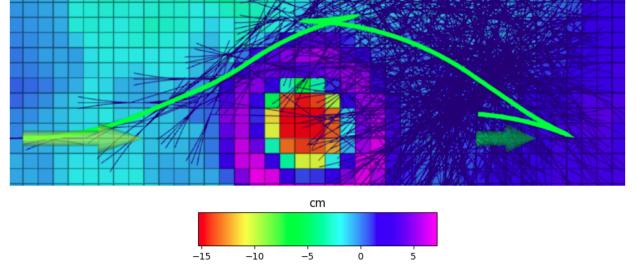


Fig. 5. The Kinematic Planner incorporates topography cost to avoid the crater with kinematically feasible solutions for the double Ackermann mobility platform. Green arrows show the starting and goal locations, the green spline shows the resulting trajectory, and the dark blue arcs show path candidates for the expanded A* lattice nodes.

The planner uses linear programming (LP) to solve the problem of minimal-energy terrain manipulation. The formulation is based on an earth mover's distance (EMD) optimization problem, which determines volumes to move between source and sink node pairs, minimizing the total sum of volume moved multiplied by the distances traveled; analogous to mechanical work ignoring transport rate.

The planar coordinates for the pairs of source and sink nodes computed in the transport plan are then used as trajectory waypoints, with the heading aligned to the minimum distance vector between the two waypoints. A third waypoint with the same heading is offset to the rear of the source node to help simplify motion planning, guiding the system to drive in and out of the crater when moving between source and sink pairs. These three waypoints are combined into a *transport triplet* ordered with the offset waypoint first, then the source node, and then the sink node.

After solving the optimization problem online, the Behavior Executive then radially orders the transport triplets and feeds the waypoints in each triplet incrementally to the Kinematic Planner (Section IV-D3) to calculate the following trajectories.

3) Kinematic Planner: The Kinematic Planner ingests *transport triplets* as goal poses from the *Transport Planner*. The goal poses from the *transport triplets* are executed until completion, designated by a Euclidean distance and orientation threshold, at which point the next transport triplet is sent to the kinematic planner for continued operation.

The Kinematic Planner includes a lattice generator, an A* search algorithm [37], and tool and velocity planners. The lattice generator creates kinematically feasible path candidates, and A* is used to find a path within position and heading thresholds of a goal pose, as seen in Figure 5. Path candidates are assigned A* cost according to weighted topography cost and cumulative distance, with a heuristic of Euclidean distance between the terminal path position to the goal. The topography cost results in path solutions that minimize climbing heights to mitigate the risk of becoming high-centered during navigation. To improve online search efficiency, the heuristic was iteratively weighted to bias the search toward finding feasible solutions within online time constraints. If the robot

deviates sufficiently from an existing plan, the kinematic planner replans a new trajectory from its current location to the existing goal pose.

The velocity planner assigns a target velocity to each waypoint along the path, depending on whether the robot is traveling forward or backward. Finally, the tool planner assigns a target blade height to each waypoint using a heuristic further explained in Section IV-E. By integrating these planners, the kinematic planner can generate optimized and feasible trajectories for the robot to execute.

E. Trajectory Control

The trajectory controller generates actuator commands for three objectives: steering, driving, and tool motion. Each of these objectives has its own specific controller. The steering controller, based on the Stanley control law [38], calculates the steer angle that can reduce and trade-off cross-track and heading error. The driving controller penalizes the driving velocity based on the steer speed and applies clamping and low-pass filters. Finally, the tool controller adjusts the tool position when the robot is either backing up or going forward to minimize undesired drag forces. The controller moves the tool down to a design height when going forward and up to an unobstructing height when backing up. This design height was set to graze flat ground when CraterGrader was placed onto the sand-filled testbed, described in Section III-E.

V. DETAILS ON OPTIMAL TRANSPORT PLANNER

The Transport Planner solves the optimization problem of minimal-energy terrain manipulation to convert an arbitrary original topography to an arbitrary design topography. The final solution minimizes the total sum of volume moved multiplied by the distances traveled, analogous to mechanical work ignoring transport rate. The optimization is formulated as two LP problems that extend on typical optimal transport problems to account for non-equal volume distributions, as described in more detail in Section V-B. Summarized in Equations (10)-(15), the two linear programs differ only by their volume constraints, depending on two cases of (1) excess sink volume, or (2) excess source volume (or equal volumes). In our application, this approach manifests as a transport plan specifying how volume-surplus regions in the worksite (referred to as *source nodes*) should be pushed to volume-lacking regions (referred to as *sink nodes*), to ultimately produce flat, graded terrain. The planner is agnostic to input type, and in this work, nodes are formed from 2.5D height map grid cells. The sets of source and sink nodes form two distributions used in optimal transport. For grading of a cratered terrain, the primary source volume is crater rims and the primary sink volume is crater floors. The final transport plan results in pushing crater rims into the crater floors.

A simple example problem is provided in Appendix Section B with Figure 10. The solution to this sample problem yields an intuitive transport plan that assists with verification of the optimal transport algorithm.

A. Key Assumptions

The planner in this paper assumes that all volumes are homogeneous material; there is no accounting for specific material types. Future work could extend this formulation by accounting for material type as an additional cost parameter. The only costs factored into this planner are the volume scales and distance to move the volumes. Importantly, the planner also makes no assumptions about how volumes will be moved, so its use is not limited to the CraterGrader platform. The final transport plan could easily be used for machines with different terraforming modalities or distributed to multiple machines.

B. Comparison to Standard Formulation

This planner extends on standard optimal transport problems by accounting for unequal distribution masses. Specifically, standard formulations either assume that distributions have the same integral areas, such as for probability distributions, or that the distributions are normalized to have the same total masses. However, inherent variability in terrain and inaccuracies in mapping mean that in practice there is nearly always unequal volume distributions. Normalizing these distributions would modify volume magnitudes which are required for scale-representative optimal transport. Maintaining these scales allows for insight into the estimated work required to execute the final plan. Allowing for unequal distribution sizes also naturally allows the planner to dictate where slack volume should remain; either “high” source volumes left untouched when all “low” sink volumes can be filled, or “low” sink volumes that should remain unfilled because all “high” source volumes are consumed. The practical application of this planner to real-world earth moving requires maintaining distribution scales and distances.

C. Optimization Objective

The optimization objective is to minimize the total volume transported between node pairs multiplied by the distances traveled. The volumes moved are continuous decision variables contained in a transport matrix $\Pi \in \mathbb{R}^{n \times m}$, where n is the number of source nodes and m is the number of sink nodes. $\Pi_{i,j}$ corresponds to the amount of volume that should be moved from the source node i to the sink node j , in node volume units. In this work, node volume units were cubic meters. Distances between nodes are calculated prior to solving the optimization problem and are contained in a distance matrix $D \in \mathbb{R}^{n \times m}$, described further in Section V-E. The objective then seeks to minimize the Frobenius inner product of Π and D . This is provided in Equation (1), with index notation used for clarity.

$$\min_{\Pi} \sum_{\substack{i=1,2,\dots,n \\ j=1,2,\dots,m}} \Pi_{i,j} D_{i,j} \quad (1)$$

To simplify the formulation, an additional constraint enforces all transport volumes to be non-negative:

$$-\Pi_{i,j} \leq 0 \quad \forall i = 1, \dots, n \text{ and } j = 1, \dots, m \quad (2)$$

D. Nodes and Volumes

The set of n source nodes is denoted \bar{Y} and the set of m sink nodes is denoted \bar{X} . Nodes are parameterized by planar worksite position coordinates (p_x, p_y) and either source volume v_y or sink volume v_x , where all volumes are defined as positive within their respective sets to simplify the optimization formulation. A source node $Y_i = \langle p_{x,i}, p_{y,i}, v_{y,i} \rangle$ is contained in the source set $\bar{Y} = \{Y_1, Y_2, \dots, Y_n\}$, where $|\bar{Y}| = n$. Similarly, a sink node $X_j = \langle p_{x,j}, p_{y,j}, v_{x,j} \rangle$ is contained in the sink set $\bar{X} = \{X_1, X_2, \dots, X_m\}$, where $|\bar{X}| = m$.

In the planner implementation, node volumes are estimated based on mapping data, where each grid cell in the map has a corresponding height. Each node volume is calculated using the grid area multiplied by the corresponding grid cell height. The grid cell area is a function of the map grid cell resolution but is a constant scalar for uniform cell sizes. The planner optimization formulation is based on volumes and assumes accounting for this calculation as data input. Note that with uniform cell sizes, a simplification could be made to use the height directly since all cells would be equally scaled by the same grid cell area.

E. Distance Matrix

A distance matrix D , defined in Equation (3), maintains the distances between planar positions of node pairs. Each element of the distance matrix $D_{i,j}$ corresponds to the Euclidean distance between source node i and sink node j , and has the same units as the node coordinate units. In this work, the distance units were meters.

$$D_{i,j} = d(Y_i, X_j) \quad \forall i = 1, \dots, n \text{ and } j = 1, \dots, m \quad (3)$$

where

$$d(Y_i, X_j) = \left\| \begin{bmatrix} p_{x,i} \\ p_{y,i} \end{bmatrix} - \begin{bmatrix} p_{x,j} \\ p_{y,j} \end{bmatrix} \right\|_2 \quad (4)$$

F. Volume Constraints

The volumes transported between nodes are constrained by the total magnitudes of source and sink volumes. In practice, these total volumes are likely to be unequal due to underlying site topography and mapping inaccuracies. To minimize mechanical work, we design constraints to allow for excess final volume from the larger of the source and sink volume magnitudes (i.e. unmoved source, or unfilled sink). Volume vectors are defined for the sets of source nodes \vec{v}_y and sink nodes \vec{v}_x in Equation (5).

$$\vec{v}_y = \begin{bmatrix} v_{y,1} \\ v_{y,2} \\ \vdots \\ v_{y,n} \end{bmatrix}_{n \times 1}, \quad \vec{v}_x = \begin{bmatrix} v_{x,1} \\ v_{x,2} \\ \vdots \\ v_{x,n} \end{bmatrix}_{m \times 1} \quad (5)$$

Consider that the relative magnitudes of source and sink volumes can only be greater than, less than, or equal to each other. These conditions can be captured in two volume cases. Note

that these cases could be captured in one unified optimization problem as a mixed integer linear program using a single binary decision variable, but for computational efficiency can instead be defined as two LP problems where only a single LP needs to be solved online after checking the volume case.

Let Case 1 be defined as when sink volume is greater than source volume (i.e., $\sum_{i=1}^n v_{y,i} < \sum_{j=1}^m v_{x,j}$). The constraints should ensure that all sink volumes can remain unfilled but are not overfilled (inequality), while all source volumes are fully used (equality):

$$\Pi^\top \vec{1}_n \leq \vec{v}_x \quad (6)$$

$$\Pi \vec{1}_m = \vec{v}_y \quad (7)$$

Similarly, let Case 2 be defined as when sink volume is the same or less than source volume (i.e., $\sum_{i=1}^n v_{y,i} \geq \sum_{j=1}^m v_{x,j}$). The constraints should then ensure that all source volumes can remain unused but are not overused (inequality), while all sink volumes are fully filled (equality):

$$\Pi \vec{1}_m \leq \vec{v}_y \quad (8)$$

$$\Pi^\top \vec{1}_n = \vec{v}_x \quad (9)$$

G. Final Optimization Problem

Combining the objective with constraints gives the following final linear programming optimization problem formulation in Equations (10)-(15). Note that this is a definition for two linear programs with identical objective (10) and transport constraint (11), where Case 1 uses additional constraints as Equations (12) - (13), and Case 2 uses additional constraints as Equations (14) - (15). The case constraints define the full respective LP, which can then be solved efficiently using any available LP solving methods. An example of solving this problem by vectorizing the decision variables is included in Appendix Section C.

$$\min_{\Pi} \quad \sum_{\substack{i=1,2,\dots,n \\ j=1,2,\dots,m}} \Pi_{i,j} D_{i,j} \quad (10)$$

$$\text{s.t.} \quad -\Pi_{i,j} \leq 0 \quad \forall i, j \quad (11)$$

$$\text{Case 1 Constraints:} \quad \left(\sum_{i=1}^n v_{y,i} < \sum_{j=1}^m v_{x,j} \right) \quad (12)$$

$$\Pi^\top \vec{1}_n \leq \vec{v}_x \quad (12)$$

$$\Pi \vec{1}_m = \vec{v}_y \quad (13)$$

$$\text{Case 2 Constraints:} \quad \left(\sum_{i=1}^n v_{y,i} \geq \sum_{j=1}^m v_{x,j} \right) \quad (14)$$

$$\Pi \vec{1}_m \leq \vec{v}_y \quad (14)$$

$$\Pi^\top \vec{1}_n = \vec{v}_x \quad (15)$$

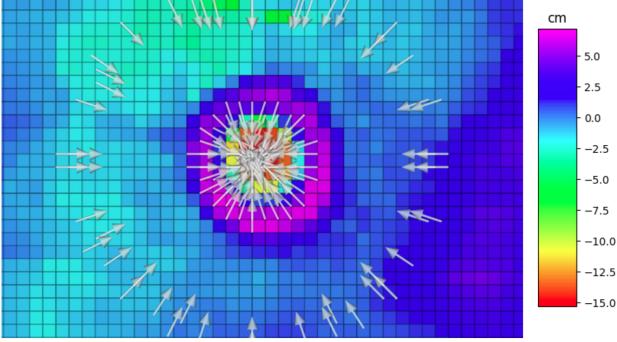


Fig. 6. The Transport Planner’s optimal online material movement solution for how to fill a real crater. In this diagram, color specifies the height of the grid cell measured from the site-wide fit plane. Red is low at -15.3 cm and purple is high at $+7.2$ cm. Each white arrow is an element of a *transport triplet*, specifying how the robot should approach the crater and manipulate terrain.

where

$$\vec{v}_y = \begin{bmatrix} v_{y,1} \\ v_{y,2} \\ \vdots \\ v_{y,n} \end{bmatrix}, \quad \vec{v}_x = \begin{bmatrix} v_{x,1} \\ v_{x,2} \\ \vdots \\ v_{x,m} \end{bmatrix}$$

H. Runtime Analysis

The overall problem size of the transport planner scales with the total number of nodes as $(n \times m)$, as computation complexity is dominated by optimizing for the continuous variables in the transport matrix $\Pi \in \mathbb{R}^{n \times m}$. Abstracted from solver specifics, the memory consumption scales with the problem size, $(n \times m)$.

The actual online runtime performance depends primarily on (1) the problem size $(n \times m)$, (2) the distribution of node volumes (which determines solving difficulty), and (3) the optimization solver used. Full solutions for ~ 1000 source and sink nodes (~ 2000 total) were observed to take approximately 30 seconds on a 2.5 GHz Quad-Core Intel Core i7 with 16 GB 1600 MHz DDR3 RAM using a C++ implementation with the GLOP solver from OR-Tools. An example of such a solution is included in Figure 9.

I. Runtime Optimizations

Implementation adjustments were made to improve online computational efficiency. Threshholding was applied to ignore source and sink nodes with volumes that were considered already close enough to the design topography. Additionally, in observing that the grading blade can push more than one cell at a time, a decimation filter was applied to the source nodes to enforce a minimum Euclidean distance between their planar coordinates. These two techniques effectively reduced the number of total nodes to speed up online computation. The threshold parameters were tuned for the anticipated crater size and grading blade width.

VI. EXPERIMENTAL RESULTS

A. Test Setup

Testing was performed at the MoonYard described in Section III-E, serving as an analog to lunar landscapes of interest for infrastructure development. Worksite terrain was initially prepared by manually smoothing to a grade within $\pm 1^\circ$, followed by digging a 1 m diameter crater near the center of the worksite, using a mold to create identical craters for repeatable testing. The crater diameter and depth-to-diameter ratio (0.2) were chosen to align within the LuSTR [2] RFP specifications. Larger craters, in both depths and diameters, could require different sensor locations and perception approaches not explored in this work.

During evaluation, CraterGrader was placed into the prepared MoonYard. Fully autonomous mapping and grading operations, described in Section III-A, were executed for an approximately 30-minute period, whereupon the robot was removed for worksite verification.

Verification was performed by capturing high-resolution point cloud scans before and after autonomous operation. A FARO Focus 3D laser scanner generated scans of worksite terrain accurate to millimeter precision, which were processed offline in an open-source 3D data analysis software. Out-of-worksit data points were removed from the point cloud and a ground plane was fitted from the remaining points. The data was then exported to an analysis pipeline to compute validation metrics such as surface grade, smoothness, and area OOS reduction.

B. Results

System performance was evaluated across five 30-minute trials of end-to-end autonomous operation, including a live demonstration. Average performance results for the four evaluation trials are shown in Table I, and results for the live performance are depicted in Table II. The key metrics to measure performance are selected as worksite grade, worksite smoothness, and area out-of-spec (OOS) reduction.

Grade and smoothness specifications are derived from the NASA LuSTR RFP [2], which define grade and smoothness requirements as $\pm 1^\circ$ from a fit plane and 1 cm in height standard deviation respectively. Figure 2 illustrates the definitions of grade angle and smoothness with respect to a crater profile. Grade is defined as the angle between a plane fit to the terrain profile, and a level plane normal to gravity. Smoothness is calculated from the standard deviation of a distribution of terrain heights relative to the fit plane. On average, CraterGrader achieved a worksite grade of 0.11° and smoothness of 0.7 cm. In a culminating live demonstration for a viewing audience, CraterGrader achieved 0.07° grade and 0.6 cm smoothness. In all cases, CraterGrader is able to repeatedly produce a smoothly graded worksite that meets the desired NASA baseline requirements in [2].

The area OOS reduction metric measures the percent reduction in worksite area that lies outside of the desired $\pm 1^\circ$ grade and 1 cm smoothness requirements before and

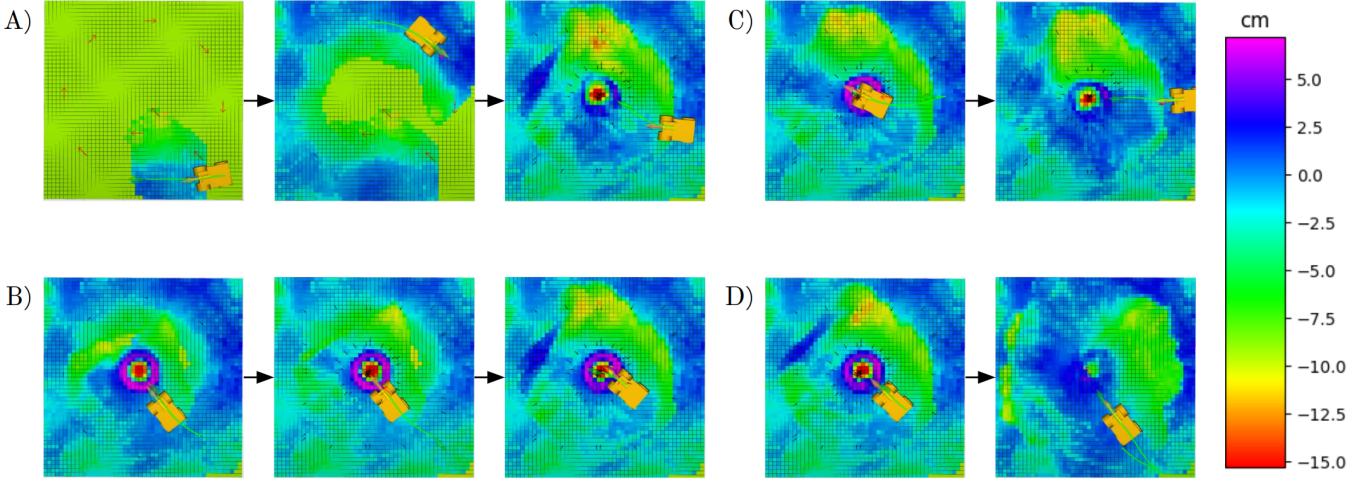


Fig. 7. Autonomous site preparation with CraterGrader begins with an **A**) exploration pass of the worksite, guided by a set of waypoints (marked as red arrows) that span the worksite. **B)** Once the map is complete, the transport planner produces *transport triplets*, (marked as black arrows) that act as waypoints dictating how the robot transports material from source nodes to sink nodes. **C)** CraterGrader reverses out of the crater to minimize difficult terrain traversal before continuing material transport. **D)** Illustrates all modifications to the site from beginning to end of operation to show how the crater feature has been predominantly cleared.

Metric	Desired	Pre-Grade	Post-Grade
Worksite Grade	$\pm 1^\circ$	0.12°	0.11°
Worksite Smoothness	< 1 cm	1.25 cm	0.7 cm
Area OOS Reduction	N/A	N/A	88%

TABLE I
CRATERGRADER AVERAGE AUTONOMOUS GRADING PERFORMANCE

Metric	Desired	Post-Grade
Worksite Grade	$\pm 1^\circ$	0.07°
Worksite Smoothness	< 1 cm	0.6 cm
Area OOS Reduction	N/A	82%

TABLE II
CRATERGRADER LIVE AUTONOMOUS GRADING PERFORMANCE

after autonomous operation. Although there are no specific requirements for area OOS reduction per the NASA LuSTR RFP [2], this metric captures the essence of CraterGrader's ability to manipulate unstructured terrain, and transform it into a worksite that meets a set of desired specifications. On average, CraterGrader achieved an 88% reduction in area OOS (from 0.45 m^2 to 0.05 m^2) for all four evaluation trials. CraterGrader also achieved an 82% reduction in area OOS (from 0.45 m^2 to 0.08 m^2) during the live demonstration, measured from the average starting conditions of the prior evaluation trials. Figure 8 illustrates an example worksite terrain topography from the evaluation trials, both pre- and post-grade, in which the prominent crater feature was flattened after autonomous operation.

C. Additional Simulation

In addition to hardware tests, additional simulations were run to demonstrate transport planner effectiveness for larger

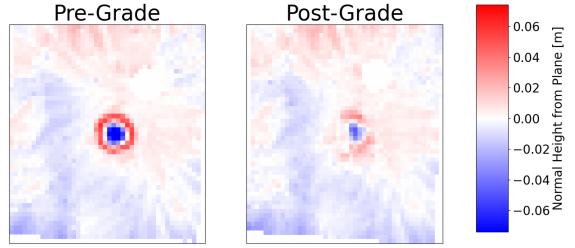


Fig. 8. Pre-Grade vs. Post-Grade verification map of a $5 \text{ m} \times 5 \text{ m}$ worksite with a single 1 m diameter crater, comparing worksite topography before and after a 30-minute fully autonomous grading operation.

maps with more direct applicability to real world scenarios. An example simulation depicting the Chandrayaan 3 landing site [39] is included in Figure 9. The worksite region used in this simulation is roughly 300 meters square. In lieu of absolute height scales (which are uncertain due to the lack of available high-resolution lunar height maps), terrain height is modeled using image pixel intensity, for the purposes of demonstrating transport planner functionality.

The simulation is designed to challenge the transport planner with a more complex terrain topography, better representative of a hypothetical lunar worksite. This expands upon the simpler terrain profiles used in hardware demonstrations. The resulting transport plan illustrates appropriate material transport from regions of high terrain to low terrain, pushing crater rims into crater floors. The simulation also shows how the transport plan seamlessly handles arbitrary terrain formations such as ridges and non-uniform craters.

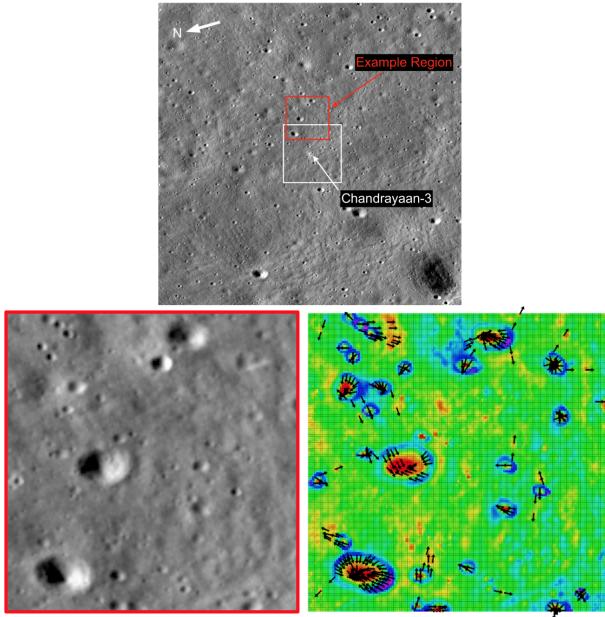


Fig. 9. Transport plan applied to a section of the Chandrayaan-3 Landing Site captured by Lunar Reconnaissance Orbiter Camera [39]. The black arrows on the right image indicate material transport.

VII. LESSONS LEARNED & KEY TAKEAWAYS

A. Patient work allows for iterative improvement with simplified hardware

- Since lunar robotics applications have a lessened focus on time efficiency compared to robustness, accuracy, and energy efficiency, it is valuable to focus on iterative operations at smaller and simpler scales. Slower, but simplified approaches can result in a more robust solution. Even in a mass-constrained system, the shallow cuts of CraterGrader's grading blade provide exceptional results due to the allowance of multiple passes and self-corrections.

B. External infrastructure provides key localization accuracy in dynamic, near-featureless environments

- There is significant value in providing external infrastructure to provide the required precision for site preparation with autonomous graders. The use of proprioceptive localization alone is limiting, especially in large environments with global accuracy requirements, due to similar lunar features, compounding error inherent in relative state sensors, and the inherent transience of site features during terrain modification. Accurate and high-precision position estimation in the z-direction is of particular importance to this application, to enable precise mapping of terrain heights and tool positioning during earthmoving operations. The robotic total station, fixed external to the worksite, resulted in highly accurate 3D positioning, independent of the feature richness in the surrounding environment.

C. Planning with a generic representation of the environment leads to flexible application and ease of error correction

- Representing the worksite as a 2.5D height map which is ingested by the high-level task planner allows for a wide variety of site states to be acted upon. This generalization is valuable for unstructured terrain modification. Additionally, CraterGrader is able to self-correct after analyzing errors in the current state of the worksite with minimal additional behavior paths. Designing the planner with explicit primitives, such as detecting craters in the worksite, would have led to more challenging recovery behaviors.

VIII. CONCLUSIONS AND FUTURE WORK

CraterGrader is a novel approach for modifying unstructured terrain to achieve a desired topography with planetary robotics constraints in mind. Unstructured and continuously deforming terrain is perceived and dynamically mapped, leveraging precision GPS-free localization subject to lunar constraints. Tool and motion planning is performed by calculating a transport plan online using a linear programming formulation of earth mover's distance and then executed with tool, drive, and steer control.

This work aims to inform future planetary site preparation autonomy development, in addition to any terrestrial applications which may benefit from similar levels of autonomy. The demonstration of our low-mass rover has shown that this approach is feasible, robust to uncertain dynamics and measurements, and well-suited to autonomous site preparation and earthmoving for a lunar environment. This represents a significant step toward the future of autonomous robots for lunar site preparation and intelligent, energy-efficient terrain manipulation in general. Future work could include introducing additional modalities of earthmoving tasks such as rock picking, compaction, drilling, blasting, etc. to create a more versatile machine. It is also in the interest of the authors to extend the work of the machine-agnostic transport planner to other heterogeneous robotic vehicles for material transport which could take the form of a scraper, bulldozer, backhoe, excavator, loader, or dump truck, possibly in multi-agent applications.

ACKNOWLEDGMENT

We would like to give special thanks to Dr. John Dolan, Warren "Chuck" Whittaker, Dr. Dimitrios Apostolopoulos, Dr. David Wettergreen, and the MRSD (Master of Science in Robotic Systems Development) program of the Carnegie Mellon University Robotics Institute for their support, opinions, and advice throughout this journey.

APPENDIX

A. Additional Media

A recording of a test run of the CraterGrader worksystem can be seen at youtube.com/watch?v=8xV1q1SpIpc.

B. Example Transport Planner Problem

Consider an example problem with source set \bar{Y} and sink set \bar{X} defined by:

$$\begin{aligned}\bar{Y} &= \{\langle p_{x,1} = -1, p_{y,1} = -0.5, v_{y,1} = 0.2 \rangle, \\ &\quad \langle p_{x,2} = 0.5, p_{y,2} = -1, v_{y,2} = 0.6 \rangle\} \\ \bar{X} &= \{\langle p_{x,1} = -2, p_{y,1} = 1, v_{x,1} = 0.3 \rangle, \\ &\quad \langle p_{x,2} = 2, p_{y,2} = 1, v_{x,2} = 0.4 \rangle\}\end{aligned}$$

In this case, there is more source volume than sink volume, as the total source volume sums to 0.8 and the total sink volume sums to 0.7. A visualization of the solution is in Figure 10. The optimal solution transports 0.2 volume from Y_1 to X_1 , 0.1 volume from Y_2 to X_1 , 0.4 volume from Y_1 to X_2 , and leaves the excess 0.1 source volume at node Y_1 . This solution minimizes the total volume moved and aligns with intuition.

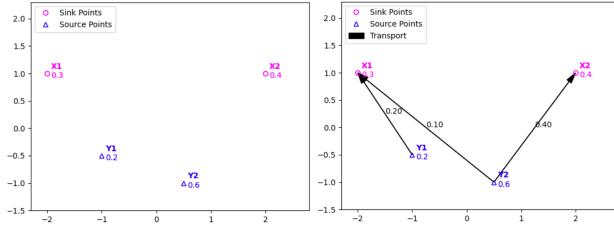


Fig. 10. Simple example problem for transport planner.

C. Transport Planner Linear Program Formulation

One way in which the transport planner optimization problem can be solved using linear programming is by converting the objective and constraints to block matrices and vectors for standard form.

$$\begin{array}{ll}\text{minimize} & c^T \hat{x} \\ \text{subject to} & G\hat{x} \leq h \\ & A\hat{x} = b\end{array}$$

To convert the objective, we need to turn the sum $\sum_{i=1,2,\dots,n} \sum_{j=1,2,\dots,m} \Pi_{i,j} D_{i,j}$ into a dot product between a vector c of constant cost terms and a vector \hat{x} of the optimization variables. The distance matrix can be converted into the constant cost vector and the transport matrix can be converted into the continuous optimization variable vector using row-major order as follows:

$$c_{nm \times 1} = \begin{bmatrix} d(Y_1, X_1) \\ d(Y_1, X_2) \\ \vdots \\ d(Y_2, X_1) \\ d(Y_2, X_2) \\ \vdots \\ d(Y_n, X_m) \end{bmatrix}, \quad \hat{x}_{nm \times 1} = \begin{bmatrix} \Pi_{1,1} \\ \Pi_{1,2} \\ \vdots \\ \Pi_{2,1} \\ \Pi_{2,2} \\ \vdots \\ \Pi_{m,n} \end{bmatrix} \quad (16)$$

To convert the constraints, we can define block matrices for creating the matrices G , A and vectors h , b for inequality and equality constraints, respectively. Note that the following formulations again assume row-major order.

To replicate the $\Pi \vec{1}_m^\top$ term, we can make a one's block matrix $B_{1,m}$ packed with the all-ones vector $\vec{1}_m^\top$ along the block diagonal. There need to be n block columns in total to have zeros applied to the rest of the optimization variable vector \hat{x} , and n block rows to sum for all n source nodes, making the final dimensions $B_{1,m} \in \mathbb{R}^{n \times (m \times n)}$.

$$B_{1,m} = \begin{bmatrix} \vec{1}_m^\top & 0 & \cdots & 0 \\ 0 & \vec{1}_m^\top & \cdots & 0 \\ \vdots & \cdots & \ddots & 0 \\ \vdots & \cdots & \cdots & \vec{1}_m^\top \end{bmatrix} \quad (17)$$

Similarly, to replicate the $\Pi^\top \vec{1}_n$ term in the matrix G , we can make a one's block matrix $B_{1,n}$ packed with n identity matrices. This then applies zeros to all of the optimization variable vector \hat{x} except for what would be each column in the transport matrix.

$$B_{1,n} = \begin{bmatrix} I_{m \times m} & I_{m \times m} & \cdots & I_{m \times m} \end{bmatrix} \quad (18)$$

These block matrices $B_{1,m}$ and $B_{1,n}$ can be used for the constraints 12, 13, 14, 15.

The last remaining constraint 11 is then to enforce that all transport volumes are positive or zero, which can be done using an $(nm \times nm)$ identity matrix I_Π with zero's vector $\mathbf{0}$ of length nm :

$$I_{nm \times nm} \hat{x} \geq \mathbf{0}_{nm \times 1} \quad (19)$$

The final LP optimization problem can then be summarized as the following:

$$\text{minimize}_{nm \times 1} c^T \hat{x} \quad (20)$$

$$\text{subject to}_{nm \times 1} G\hat{x} \leq h \quad (21)$$

$$A\hat{x} = b \quad (22)$$

where for Case 1 when there is more sink than source ($\sum_{i=1}^n v_{y,i} < \sum_{j=1}^m v_{x,j}$), we have:

$$G_{(nm+m) \times nm} = \begin{bmatrix} -I_\Pi \\ B_{1,n} \end{bmatrix} \quad (23)$$

$$h_{(nm+m) \times 1} = \begin{bmatrix} \mathbf{0} \\ \vec{v}_x \end{bmatrix} \quad (24)$$

$$A_{n \times nm} = [B_{1,m}] \quad (25)$$

$$b_{n \times 1} = [\vec{v}_y] \quad (26)$$

while for Case 2 when there is more source than sink, or equal amounts, $(\sum_{i=1}^n v_{y,i} \geq \sum_{j=1}^m v_{x,j})$, we have:

$$G_{(nm+n) \times nm} = \begin{bmatrix} -I_{\Pi} \\ B_{1,m} \end{bmatrix} \quad (27)$$

$$h_{(nm+n) \times 1} = \begin{bmatrix} \mathbf{0} \\ \vec{v}_y \end{bmatrix} \quad (28)$$

$$A_{m \times nm} = [B_{1,n}] \quad (29)$$

$$b_{m \times 1} = [\vec{v}_x] \quad (30)$$

The final transport solution is then contained in the optimization variable vector \hat{x} , which can be converted back to the full transport matrix Π by reshaping it to $n \times m$ using row-major order.

REFERENCES

- [1] NASA's Plan for Sustained Lunar Exploration and Development, 2020.
- [2] NASA. Space Technology Research Grants Program, Lunar Surface Technology Research Opportunities Appendix. NASA NSpires, 2021.
- [3] K Skonieczny, DS Wettergreen, and WL Whittaker. Advantages of continuous excavation in lightweight planetary robotic operations. *The International Journal of Robotics Research*, 35(9):1121–1139, 2016. doi: 10.1177/0278364915615689.
- [4] David P Miller and Kyle Machulis. Visual aids for lunar rover tele-operation. In *proceedings of 8th International Symposium on Artificial Intelligence, Robotics and Automation in Space*, edited by R. Battick, ESA Publishing, Noordwijk, Netherlands, 2005.
- [5] Nathan Melenbrink, Justin Werfel, and Achim Menges. On-site autonomous construction robots: Towards unsupervised building. *Automation in Construction*, 119: 103312, 2020. ISSN 0926-5805. doi: <https://doi.org/10.1016/j.autcon.2020.103312>.
- [6] Sung-Keun Kim and Jeffrey S Russell. Framework for an intelligent earthwork system: Part I. System architecture. *Automation in Construction*, 12(1):1–13, 2003. ISSN 0926-5805. doi: [https://doi.org/10.1016/S0926-5805\(02\)00034-1](https://doi.org/10.1016/S0926-5805(02)00034-1).
- [7] Adrian Bonchis, Nicholas Hillier, Julian Ryde, Elliot Duff, and Cédric Pradalier. Experiments in Autonomous Earth Moving. *IFAC Proceedings Volumes*, 44(1):11588–11593, 2011. ISSN 1474-6670. doi: <https://doi.org/10.3182/20110828-6-IT-1002.00536>. 18th IFAC World Congress.
- [8] Joshua A. Marshall, Patrick F. Murphy, and Laeeque K. Daneshmend. Toward autonomous excavation of fragmented rock: Full-scale experiments. *IEEE Transactions on Automation Science and Engineering*, 5(3):562 – 566, 2008. doi: 10.1109/TASE.2007.912709.
- [9] Daniel Schmidt, Martin Proetzsch, and Karsten Berns. Simulation and control of an autonomous bucket excavator for landscaping tasks. In *2010 IEEE International Conference on Robotics and Automation*, pages 5108–5113, 2010. doi: 10.1109/ROBOT.2010.5509546.
- [10] Dominic Jud, Simon Kerscher, Martin Wermelinger, Edo Jelavic, Pascal Egli, Philipp Leemann, Gabriel Hottiger, and Marco Hutter. HEAP - The autonomous walking excavator. *Automation in Construction*, 129:103783, 2021. ISSN 0926-5805. doi: <https://doi.org/10.1016/j.autcon.2021.103783>.
- [11] Caterpillar. Cat Grade with 3D for Dozers, 2023.
- [12] Built Robotics. Everest, Autonomy at Your Fingertips, 2023.
- [13] T Muff, L Johnson, R King, and MB Duke. A prototype bucket wheel excavator for the Moon, Mars and Phobos. In *AIP conference proceedings*, volume 699, pages 967–974. American Institute of Physics, 2004.
- [14] JM Schuler, JD Smith, RP Mueller, and AJ Nick. RASSOR, the Reduced Gravity Excavator. *Lunar ISRU 2019, Developing a New Space Economy through Lunar Resources and Their Utilization*, 5061, abstract, 2019.
- [15] Robert Mueller, Jason Schuler, Andrew Nick, Allen Wilkinson, Christopher Gallo, and Robert King. Lightweight bulldozer attachment for construction and excavation on the lunar surface. In *AIAA SPACE 2009 conference & exposition*, page 6466, 2009.
- [16] Krzysztof Skonieczny, DS Wettergreen, and WL “Red” Whittaker. Parameters governing regolith site work by small robots. In *Earth and Space 2010: Engineering, Science, Construction, and Operations in Challenging Environments*, pages 1326–1333. 2010.
- [17] Krzysztof Skonieczny, Matthew Delaney, David Wettergreen, and William Whittaker. Productive Lightweight Robotic Excavation for the Moon and Mars. *Journal of Aerospace Engineering*, 27, 07 2014. doi: 10.1061/(ASCE)AS.1943-5525.0000342.
- [18] RM Kelso, R Romo, C Andersen, RP Mueller, T Lippitt, NJ Gelino, JD Smith, Ivan I Townsend, JM Schuler, MW Nugent, et al. Planetary basalt field project: Construction of a lunar launch/landing pad, PISCES and NASA Kennedy space center project update. In *Earth and Space 2016: Engineering for Extreme Environments*, pages 653–667. American Society of Civil Engineers Reston, VA, 2016.
- [19] Jekanthan Thangavelautham, Kenneth Law, Terence Fu, Nader Abu El Samid, Alexander DS Smith, and Gabriele MT D’Eleuterio. Autonomous multirobot excavation for lunar applications. *Robotica*, 35(12):2330–2362, 2017.
- [20] Yang Cheng, Mark Maimone, and Larry Matthies. Visual odometry on the Mars exploration rovers. In *2005 IEEE International Conference on Systems, Man and Cybernetics*, volume 1, pages 903–910. IEEE, 2005.
- [21] Hong Jiang, Xiaoying Sun, and Xuezhi Yan. Lunar Rover Positioning based on Time of Arrival Measurements of UWB Signals. In *2008 4th International Conference on Wireless Communications, Networking and Mobile Computing*, pages 1–4. IEEE, 2008.

- [22] Ben Wu, Philippe Ludivig, Ross Potter, Andrew Chung, and Timothy Seabrook. ABSOLUTE LOCALIZATION FOR SURFACE ROBOTICS IN GPS-DENIED ENVIRONMENTS USING A NEURAL NETWORK. In *International Symposium on Artificial Intelligence, Robotics and Automation in Space*, 2020, 2020.
- [23] Larry Matthies, Shreyansh Daftry, Scott Tepsuporn, Yang Cheng, Deegan Atha, R. Michael Swan, Sanjna Ravichandar, and Masahiro Ono. Lunar Rover Localization Using Craters as Landmarks. In *2022 IEEE Aerospace Conference (AERO)*, pages 1–17, 2022. doi: 10.1109/AERO53065.2022.9843714.
- [24] Pablo Molina, Peter Iles, and Kirk MacTavish. LANDER-BASED LOCALIZATION SYSTEM OF THE LUNAR ANALOGUE ROVER “ARTEMIS”. In *International Symposium on Artificial Intelligence, Robotics and Automation in Space*, pages 4–6, 2012.
- [25] David Wettergreen, Scott Moreland, Krzysztof Skonieczny, Dominic Jonak, David Kohanbash, and James Teza. Design and field experimentation of a prototype lunar prospector. *The International journal of robotics research*, 29(12):1550–1564, 2010.
- [26] Gaspard Monge. Mémoire sur la théorie des déblais et des remblais. *Mem. Math. Phys. Acad. Royale Sci.*, pages 666–704, 1781.
- [27] Martin Burger, Ina Humpert, and Jan-Frederik Pietschmann. Dynamic optimal transport on networks, 2021.
- [28] Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. The earth mover’s distance as a metric for image retrieval. *International journal of computer vision*, 40(2):99, 2000.
- [29] Justin Solomon, Raif Rustamov, Leonidas Guibas, and Adrian Butscher. Earth mover’s distances on discrete surfaces. *ACM Transactions on Graphics (ToG)*, 33(4):1–12, 2014.
- [30] Carsten Gottschlich and Stephan Huckemann. Separating the real from the synthetic: minutiae histograms as fingerprints of fingerprints. *IET Biometrics*, 3(4):291–301, 2014.
- [31] Elisa Ricci, Gloria Zen, Nicu Sebe, and Stefano Meselodi. A Prototype Learning Framework Using EMD: Application to Complex Scenes Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(3):513–526, 2013. doi: 10.1109/TPAMI.2012.131.
- [32] Haoyan Zhai, Magnus Egerstedt, and Haomin Zhou. Path Planning in Unknown Environments Using Optimal Transport Theory, 2019.
- [33] Steven Garlinge, David Mares, and Daniel O’Shaughnessy. 3D Positioning Systems for Underground Construction Robots. In *Association of Public Authority Surveyors Webinar Series*. APAS, 2020.
- [34] Paul Furgale, John Enright, and Timothy Barfoot. Sun Sensor Navigation for Planetary Rovers: Theory and Field Testing. *IEEE Transactions on Aerospace and Electronic Systems*, 47(3):1631–1647, 2011. doi: 10.1109/TAES.2011.5937255.
- [35] Thomas Moore and Daniel Stouch. A Generalized Extended Kalman Filter Implementation for the Robot Operating System. In Emanuele Menegatti, Nathan Michael, Karsten Berns, and Hiroaki Yamaguchi, editors, *Intelligent Autonomous Systems 13*, pages 335–348, Cham, 2016. Springer International Publishing. ISBN 978-3-319-08338-4.
- [36] Lars B Cremean and Richard M Murray. Uncertainty-based sensor fusion of range data for real-time digital elevation mapping (rtdem). In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 18–22, 2005.
- [37] Peter Hart, Nils Nilsson, and Bertram Raphael. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968. doi: 10.1109/tssc.1968.300136.
- [38] Gabriel M Hoffmann, Claire J Tomlin, Michael Montemerlo, and Sebastian Thrun. Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing. In *2007 American control conference*, pages 2296–2301. IEEE, 2007.
- [39] Mark Robinson. Chandrayaan-3 Landing Site, 2023.