18th International Conference on Space Operations, Montreal, Canada, 26 - 30 May 2025.
Copyright 2025 by the Canadian Space Agency (CSA) on behalf of SpaceOps. All rights reserved. One or more authors of this work are employees of the government of the United States of America, which may preclude the work from being subject to copyright in United States of America, in which event no copyright is asserted in that country.

SpaceOps-2025, ID #559

SHERPA — An Artificial Intelligence System for Mission Planning and Decision Support

Edward Balaban[a]*, Zachary Booth[b], Evan Astle[c], Michelle Ho[d], Hailey Warner[e], Grace Kim[f], Duncan Eddy[g], Somrita Banerjee[h], Joshua Ott[i], Mark Shirley[j], Kevin Bradner[k], Anthony Colaprete[l]

[a] *NASA Ames Research Center, United States of America*, edward.balaban@nasa.gov
[b] *KBR Inc. at NASA Ames Research Center, United States of America*, zachary.m.booth@nasa.gov
[c] *KBR Inc. at NASA Ames Research Center, United States of America*, david.e.astle@nasa.gov
[d] *KBR Inc. at NASA Ames Research Center, United States of America*, michelle.t.ho@nasa.gov
[e] *KBR Inc. at NASA Ames Research Center, United States of America*, hailey.l.warner@nasa.gov
[f] *KBR Inc. at NASA Ames Research Center, United States of America*, grace.r.kim@nasa.gov
[g] *Department of Aeronautics & Astronautics, Stanford University, United States of America*, deddy@stanford.edu
[h] *KBR Inc. at NASA Ames Research Center, United States of America*, somrita.banerjee@nasa.gov
[i] *Department of Aeronautics & Astronautics, Stanford University, United States of America*, joshuaott@stanford.edu
[j] *NASA Ames Research Center, United States of America*, mark.h.shirley@nasa.gov
[k] *NASA Ames Research Center, United States of America*, kevin.bradner@nasa.gov
[l] *NASA Ames Research Center, United States of America*, anthony.colaprete-1@nasa.gov

* Corresponding Author

Abstract

SHERPA (System Health Enabled Real-time Planning Advisor) is an artificial intelligence decision support system for space missions based on methods for formal decision making under uncertainty. Among its capabilities, it supports planning for vehicles/systems with degrading or faulty components. SHERPA is currently serving as the primary strategic mission planning system on the Volatiles Investigating Polar Exploration Rover (VIPER) mission. On VIPER, SHERPA has been used in selecting the mission area, determining the landing site, performing system engineering studies, and generating strategic mission plans. During the mission, SHERPA will be used to make strategic plan adjustments based on the observed rover performance and scientific data collected up to that point. While the development of SHERPA has been motivated and funded by the VIPER mission (and its predecessor, Resource Prospector), SHERPA is architected to be modular, extensible, and applicable to a wide range of space missions — both robotic and crewed. The main reasoning algorithms currently used by SHERPA are based on Markov decision processes, MDPs, and partially observable Markov decision processes, POMDPs. The paper describes SHERPA's architecture and capabilities, details its use on VIPER, presents the results achieved to date, finishing with a discussion of the current work on enhancing SHERPA's capabilities and deploying it on other missions.
Keywords: AI, planning, uncertainty, MDP, POMDP

Nomenclature

s	State
b	Belief state
a	Action
o	Observation
r	Reward
S	State space
B	Belief space
A	Action space
O	Observation space
T	State transition function
Z	Observation function
R	Reward function

Acronyms/Abbreviations

AI	Artificial intelligence
MDP	Markov decision process
POMDP	Partially observable Markov decision process

Copyright 2025 by the Canadian Space Agency (CSA) on behalf of SpaceOps. All rights reserved. One or more authors of this work are employees of the government of the United States of America, which may preclude the work from being subject to copyright in United States of America, in which event no copyright is asserted in that country.

SHERPA	System Health Enabled Realtime Planning Advisor
VIPER	Volatiles Investigating Polar Exploration Rover
RPA	Robust Precomputed Autonomy
MCTS	Monte-Carlo Tree Search
SARSOP	Successive Approximations of the Reachable Space under Optimal Policies
PSR	Permanently shadowed region

1. Introduction

Space missions are among the most challenging of human endeavors, due to the harshness and the inherent uncertainties of the operating environment, as well as the complexity of the technologies involved. From the early days of space exploration until today, operating space missions has typically required large teams of experts performing detailed planning and making real-time operational decisions. While automation tools introduced over the last several decades have helped alleviate the workload of human operators, analyzing numerous potential mission scenarios and contingencies while pulling together information from heterogeneous sources is still often a time-consuming and error-prone task. As Gaines et al illustrate on the example of the Mars Curiosity rover mission [1], time expended on analyzing the vehicle state and defining next sets of activities has resulted in productivity challenges and underutilization of the vehicle.

This paper describes an artificial intelligence (AI) decision support system for space missions developed on the foundation of formal methods for decision making under uncertainty — System Health Enabled Realtime Planning Adviser (SHERPA). SHERPA was created with the goal of further reducing the burden on mission planners and operators by reasoning through and learning from mission scenarios where uncertainty in state estimation and action outcomes may be present, then recommending an optimized mission plan with built-in contingency branches. SHERPA's first application is NASA’s Volatiles Investigating Polar Exploration Rover (VIPER) mission [2]. It is also currently being used in the development of a number of other space missions.

In the rest of this section, we overview prior and related work, as well as introduce the VIPER mission, which is used throughout the paper to illustrate the current capabilities of SHERPA. Section 2 provides a brief overview of decision making under uncertainty. Section 3 describes the overall architecture of SHERPA, followed by examples of its use cases and algorithms presented in Section 4. Section 5 goes into more detail on SHERPA’s use on the VIPER mission. Section 6 outlines the current work to enhance SHERPA’s capabilities and deploy it on additional missions. Section 7 concludes the paper.

1.1 Prior and related work

SHERPA traces its roots to the mission planning and system health management work conducted at NASA Ames Research Center from the late 1990s to the mid-2010s. The Remote Agent system, demonstrated during the 1998-2001 Deep Space 1 mission, included a planning component and also pioneered the approach of using information about system health and operational constraints to make execution decisions [3]. The latter part was based on an AI-based reasoner called Livingstone.

The next version of Livingstone, Livingstone 2, was then used in developing a fault diagnosis and recovery system for the propulsion component of the X-34 experimental reusable space plane [4]. It was also part of the autonomy package on the Personal Satellite Assistant (PSA) robot [5], intended to help with diagnosing issues aboard the International Space Station and developing mitigation recommendations. A different AI reasoner, Hybrid Diagnosis Engine (HyDE) [6] was later used for a similar purpose — to provide fault diagnosis and recovery capability for an autonomous planetary drill prototype [7].

Transition to providing broader mission management capabilities took place in support of work to optimize unmanned aerial vehicle operations [8]. The SHERPA project started in 2015 with seed funding from the NASA Ames Director’s Discretionary Fund and the goal of demonstrating a prototype decision support system [9] in application to the Resource Prospector mission concept [10]. When the VIPER mission was formulated on the basis of Resource Prospector, SHERPA became part of its mission planning pipeline, eventually becoming the primary strategic planning system for the mission.

A number of other automated activity and path planning systems have been used on space missions in the past. One notable example is EUROPA, a constraint satisfaction planner that has been used on multiple NASA missions [11]. In particular, EUROPA was used as the core reasoning algorithm within the Mixed-initiative Activity Plan GENerator (MAPGEN) for the Mars Exploration Rovers mission [12]. It was also integrated into Ensemble, a ground operations tool used on NASA's Phoenix Lander and Mars Science Laboratory missions [13]. More recently EUROPA was used in SPIFe, a software environment for creating activity plans, as well as in its successor OpenSPIFe, described in [14].

Copyright 2025 by the Canadian Space Agency (CSA) on behalf of SpaceOps. All rights reserved. One or more authors of this work are employees of the government of the United States of America, which may preclude the work from being subject to copyright in United States of America, in which event no copyright is asserted in that country.

The AutoNav system created at JPL on a variant of the A* algorithm to automatically generate traverse segments for Mars rovers [15]. SHERPA is distinguished from the examples above by reasoning over probabilistic models of action outcomes and state uncertainty (while pursuing optimized solutions), rather than making deterministic assumptions or relying on margins defined a priori. Additionally, SHERPA is capable of planning for systems with degrading or faulty components.

1.2 The VIPER mission

The VIPER mission is a lunar polar volatiles prospecting mission developed through NASA's Science Mission Directorate (Planetary Science Division). The mission includes a rover-borne payload that (1) can locate surface and near-subsurface volatiles, (2) excavate and analyse samples of the volatile-bearing regolith, and (3) demonstrate the form, extractability and usefulness of the materials. VIPER's primary mission goal is to characterize the distribution of water and volatiles across a range of thermal environments. The VIPER science payload (Figure 1) consists of a near-infrared spectrometer (Near-InfraRed Spectrometer System, NIRVSS), neutron spectrometer (Neutron Spectrometer System, NSS), mass spectrometer (Mass Spectrometer for Observing Lunar Operations, MSOLO), and subsurface drill (The Regolith and Ice Drill for Exploring New Terrain, TRIDENT). The rover's navigation cameras and the inertial measurement unit (IMU) will also used for scientific data collection.

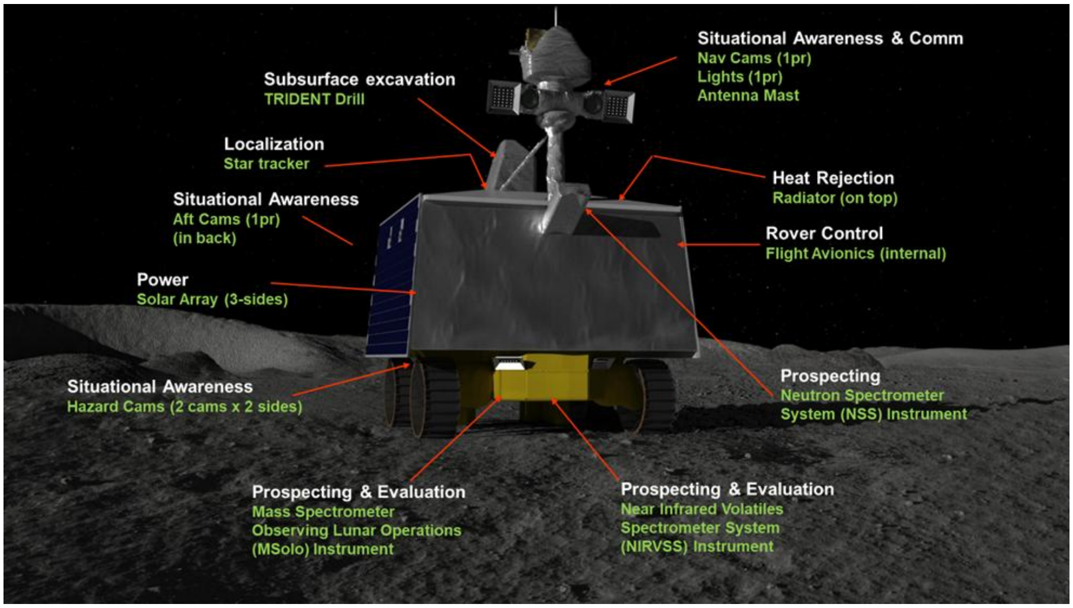

Figure 1. The VIPER rover and its navigation, prospecting, surface and subsurface assay science instruments.

The rover is solar-powered and has a Li-Ion battery for energy storage. The battery allows the rover to operate for roughly ten hours in shadow (in the scientifically interesting permanently shadowed regions, for instance). The rover will be teleoperated from Earth, with limited on-board autonomous capability. It is, therefore, critical for the rover to stay within the line of sight to one of the Deep Space Network (DSN) ground stations while operating. When the Earth is below the lunar horizon and no DSN stations are visible, the rover must be parked in a location (safe haven) where the duration of continuous sun shadow does not exceed the rover's survival capability in low-power hibernation mode (a maximum of 50 hours). Another important consideration is the steepness of slopes that the rover navigates. For planning purposes, the slopes are constrained to a maximum of 15 degrees.

In lunar polar regions, both Sun and Earth never rise high above the horizon, thus resulting in large, fast-moving illumination and communication shadows created by local topographical features. Keeping the rover out of these shadows (communication shadows in particular) and getting it to a safe haven before the Earth sets below the horizon (for roughly two weeks each month), are the most important mission planning tasks. The VIPER mission is planned to last 5-6 Earth months during the lunar polar "spring" and "summer" seasons, when there is a sufficient amount of sunlight in the planned mission area (on a large plateau on the top of Mons Mouton).

Copyright 2025 by the Canadian Space Agency (CSA) on behalf of SpaceOps. All rights reserved. One or more authors of this work are employees of the government of the United States of America, which may preclude the work from being subject to copyright in United States of America, in which event no copyright is asserted in that country.

2. Decision making under uncertainty

The overall artificial intelligence approach adopted in implementing SHERPA is *sequential decision making under uncertainty (sequential DMU)*. This section briefly overviews the general concepts of sequential DMU as they apply to SHERPA. Notation in this section generally follows standard sequential DMU conventions [16].

A system and its operating environment (the world) may assume a number of distinct configurations (for realistic systems this number is typically infinite). A state s is a descriptor of such a configuration. In this work, states are assumed to have the Markov property [17], i.e., that a state s of a system is defined in such a way that the next state, s', depends only on s and not on any previous states. All possible states of a given system form its state space S.
An *action* initiates transition from the current state s to the next state s'. All available control actions for a system form its *action space* A (A may be state-dependent).

It is not always possible to determine the state of a system and its operating environment exactly. An *observation* o is information about the state, even if indirect, that is possible to obtain. Observations can be obtained through system sensors, for example, and in many cases may be incomplete, noisy, or even contradictory. All observations possible for a system form its *observation space* O.

A *model* is an abstracted representation of a system and its operating environment (or a representation of a particular aspect of them). Models may be represented as functions that take variable inputs and generate an output. A model may be deterministic or stochastic; in the former case the model always produces the same output for the same input, in the latter case the output is stochastic. In this work, most of the models used are stochastic.

A state transition model takes the form $T(s'|s,a)$, describing the probability of transitioning to a particular state s' as a result of taking action a from state s. An observation model $Z(o|s,a,s')$ describes the probability of getting an observation o upon transition to state s' from state s as a result of action a. A reward model $R(s,a)$ describes a reward obtained as a result of reaching state s' by taking action a in state s. R is the only fundamental model type used in this work that is deterministic, as stochastic reward models have been shown to provide no benefits for the decision-making methods used.

If executing action a in state s for some system is not guaranteed to transition the system to a unique next state s', the system is defined to have *action outcome uncertainty* (or, equivalently, *outcome uncertainty* for brevity). If modeling outcome uncertainty is important, a stochastic state transition model is used.

When the current state s of a system cannot be determined exactly, the system is defined to have *state uncertainty*. If observations are available that contain at least some information about the true system state (and thus may help reduce state uncertainty), the states are defined to have *partial observability*. In modeling systems with partial state observability for decision-making purposes, *belief states* may be used. A belief state (or *belief*, for brevity) captures the information available about a partially observable true system state. A belief b may be modeled as a probability distribution over S, with B denoting the space of all beliefs.

Along with state-space models, utility theory [18] is fundamental to our work on SHERPA. *Utility* is a numerical measure of preference over the space of possible outcomes. A related concept is that of a *utility function* (or, equivalently, *value function*), that defines such a numerical measure for a particular set of input variables. For instance, if a utility function is defined for lunar rover states, a higher utility value may be assigned to a state where a desired scientific location has been reached and a lower value to a state where a malfunction has occurred.

The general function of decision making is to select actions. While in some systems the focus of decision making is only on choosing a single, immediate a_t at a given time t, real-world decision-making problems often involve considering a sequence of actions. The length of the sequence, i.e., how many time intervals or decision-making steps are being looked ahead, is defined as the *decision horizon* (or, equivalently, *planning horizon*). In a strictly deterministic system operating in a deterministic environment, an entire sequence of actions — a *plan* — can be selected ahead of time. In systems with action outcome uncertainty, however, a fixed plan can quickly become obsolete. Instead, a *policy* $p(s): S \rightarrow A$ needs to be selected that prescribes which action is to be taken in any state. In a partially observable setting, a policy maps beliefs to actions, i.e., $p(b): B \rightarrow A$.

A policy can be either *offline* (precomputed for all states or beliefs of interest) or *online* (computed for the current state or belief only). An *optimal policy* p^* is a policy that, in expectation, optimizes a desired metric. In our work, the metric optimized is state (or belief state) utility. Problems with outcome uncertainty are often modeled as Markov decision processes (MDPs) and problems with both outcome and state uncertainty are typically modeled as partially observable Markov decision processes (POMDPs).

3. SHERPA architecture

The high-level architecture of SHERPA is depicted in Figure 2. From the start of the development effort, the architecture was intended to be modular, extensible, and model-based. SHERPA is organized around use cases, each

Copyright 2025 by the Canadian Space Agency (CSA) on behalf of SpaceOps. All rights reserved. One or more authors of this work are employees of the government of the United States of America, which may preclude the work from being subject to copyright in United States of America, in which event no copyright is asserted in that country.

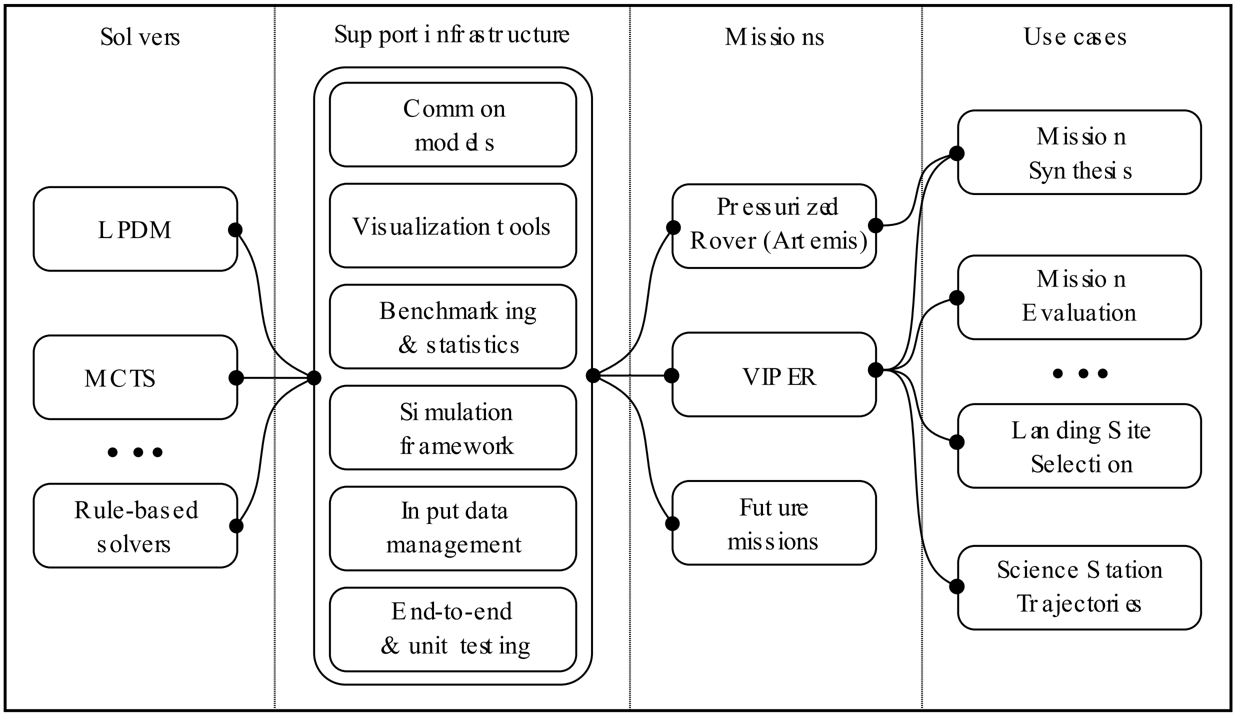

Figure 2. SHERPA architecture.

representing a particular type of a decision-making problem, with examples including landing site selection, mission synthesis, and mission evaluation. The use cases are designed for maximum code reuse. Ideally, only the models of the mission, the vehicle, and the operating environment would need to be replaced to support a new mission. In practice, some code adaptation has been required so far to extend to new missions, but we expect that with each additional application the generality of the use cases will continue to increase, thus over time minimizing the amount of code adaptation required.

Most current use cases are formulated either as MDPs or POMDPs, describing the model elements introduced in the previous section. A mission can be associated with several use cases, and the models of the mission, the vehicle, and the environment often shared among them. Some use cases are created by incorporating other use cases. For instance, for the VIPER mission, the Landing Site Selection use case is based on a high-level reasoning loop that calls Mission Synthesis and Mission Evaluation.

In addition to the use case infrastructure, SHERPA provides a library of graphical user interface components that can be assembled into a user interface best suited to a particular mission. These components include 2-D and 3-D map panes, system state displays, mission timeline viewers, graph modules, execution control modules, mission status displays, and others. SHERPA also provides facilities for input data management, such as map sets, activity dictionaries, and user-specified mission plan preferences.

SHERPA is implemented in the Julia programming language [19]. Julia is a modern, high-performance language for scientific computing with native support for parallel processing. SHERPA also incorporates support for the POMDPs.jl [20] application programming interface, which provides compatibility with many community-developed decision-making algorithms (and support tools) created using Julia and allows to select the most appropriate algorithm for a specific problem. For instance, the algorithm used for VIPER's strategic mission planning is an open-source Julia implementation of Monte Carlo Tree Search, or MCTS [21].

4. Key use cases and algorithms

This section overviews the key SHERPA use cases and the algorithms used to implement them. These use cases are general enough to be applicable not just to VIPER, but to other similar missions. Use cases specific to VIPER (such as the use case that generates detailed rover trajectories and drill site placements within VIPER's science stations) are left for future publications.

Copyright 2025 by the Canadian Space Agency (CSA) on behalf of SpaceOps. All rights reserved. One or more authors of this work are employees of the government of the United States of America, which may preclude the work from being subject to copyright in United States of America, in which event no copyright is asserted in that country.

4.1 Mission Synthesis

The Mission Synthesis use case automatically generates optimized plans for a user specified set of starting conditions and model parameter settings. This includes finding a baseline plan (Figure 3) that maximizes science return (while accounting for vehicle health and robustness to sun shadow and DSN coverage), then identifying weak (bottleneck) points within the baseline plan and creating contingency plans to further mitigate the most probable risks (Figure 4).

Mission Composer is the primary entry point into Mission Synthesis for the users. From the user's provided mission scenario, it prompts Segment Generation to find plan segments for any novel states that have not been computed previously. It then delegates to Robust Decision Making (RDM) to find the best baseline plan for the scenario, which is immediately available to the user for further iteration. Once the user is satisfied with the baseline plan, Mission Composer can, optionally, send it to Bottleneck Identification to find a list of states where the mission is most at risk. It then iterates through the list and finds optimal contingency plans that appropriately mitigate the identified risk at that state.

Finally, Mission Composer combines the baseline plan with the contingency plans to form a full mission plan, which is again made available to the user for further iteration and refinement. Both the baseline and the full mission plans can be scored via the Mission Evaluation use case, which Mission Composer can optionally call automatically when it finishes generating a plan.

Robust Decision Making leverages Segment Generation, an MDP/POMDP solver (MCTS in the case of VIPER) and detailed operations simulations to discover and score potential plans iteratively. Given a starting state (typically by Mission Composer), RDM will look up the possible plan segments which start at that state and choose one of them as an action. It will then simulate operations for that plan segment, and accumulate reward based off the stochastic simulation. If the simulation reaches the end of the plan segment, the state transition model will adopt the final state in the plan as the starting state for the next cycle of the simulation, and continue in this fashion until either the simulation or the underlying plan reaches a terminal state. Then the accumulated rewards are aggregated up the policy tree and used to inform subsequent simulations. The final plan is an ordered list of segments that score best once the model has converged.

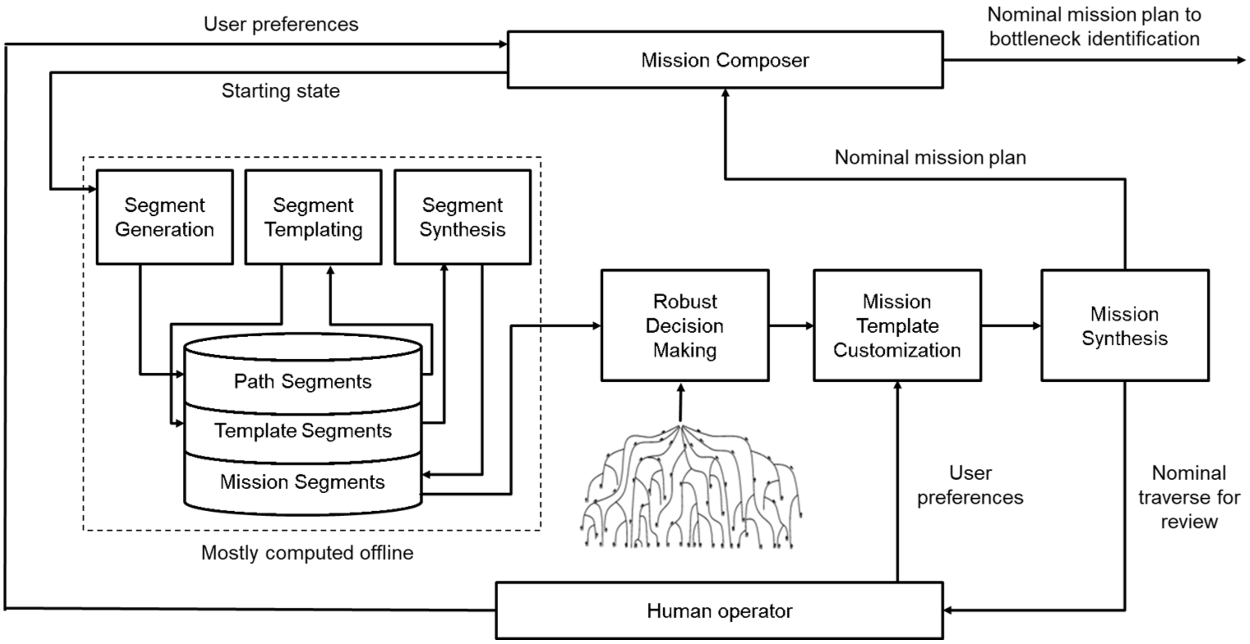

Figure 3. The algorithm for generating nominal mission plans.

Segment Generation supplies RDM with segments by iterating over a list of starting states and finding segments to all the mission sites of interest. Segment Generation is divided into Segment Outlining, Segment Templating, Segment Synthesis, and a segment store. Segment Outlining is responsible for finding robust paths between starting states and sites of interest, accounting for sun shadow and DSN coverage. Segment Templating is responsible for finding a set of

Copyright 2025 by the Canadian Space Agency (CSA) on behalf of SpaceOps. All rights reserved. One or more authors of this work are employees of the government of the United States of America, which may preclude the work from being subject to copyright in United States of America, in which event no copyright is asserted in that country.

target placements for each of these paths, where each placement in the set is optimal based off the knowledge of what mission objectives have been completed in prior segments.

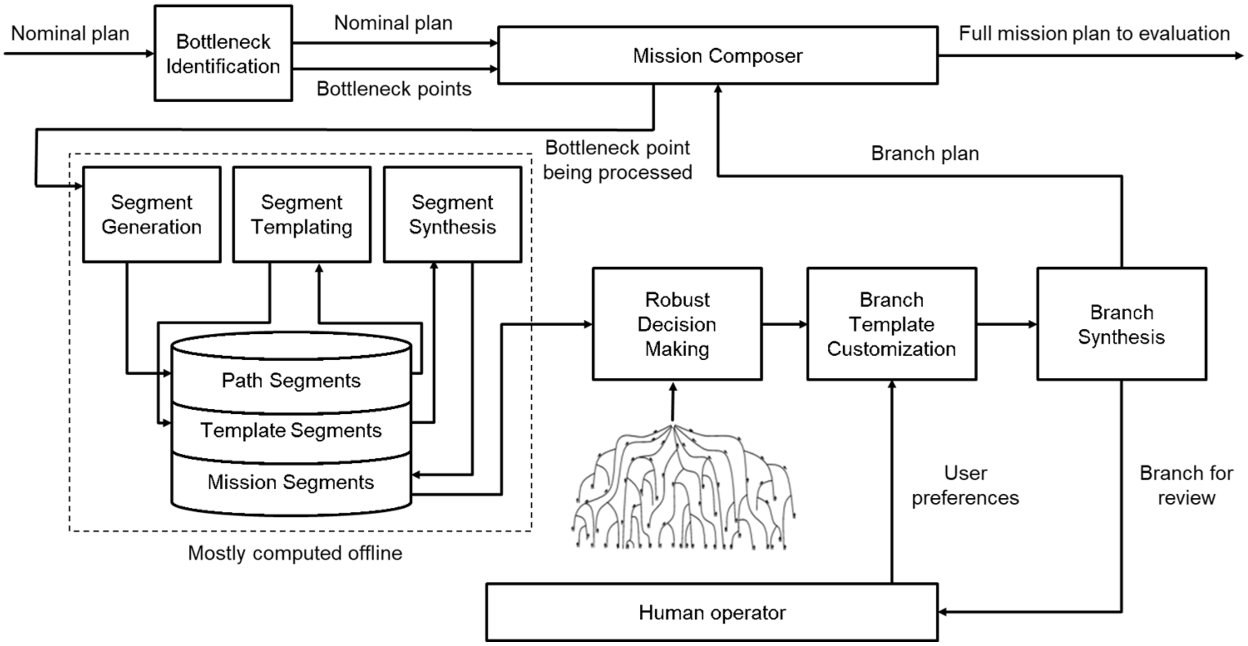

Figure 4. The algorithm for generating contingency branches.

Template segments are then synthesized to find precise timings and accurate final states for the segment. Each plan segment is stored in the segment store for future use, indexed by its initial state, target action, and parameters used to generate it. If any non-terminal final states generated during Segment Synthesis are novel, they are pushed into the starting states queue for Segment Outlining and processed — until no unprocessed states are found.

Segment Generation is intended to be run both online and offline. Offline generation is appropriate for inferable starting states, like landing sites or existing safe havens, and is intended to cut down on necessary and repetitive computations while exploring scenarios. Online generation is used by RDM to fill in any gaps in the existing generated store, and uses the same interface regardless of whether the state is novel or already explored.

4.2 Mission Evaluation

A mission plan generated by Mission Synthesis can be stress-tested through stochastic mission simulations, where variability in key mission quantities (such as the vehicle speed, activity durations, and power consumption) is introduced (along with off-nominal events, such as communication dropouts). SHERPA employs action policies for off-nominal situations mimicking the expected responses of a trained human operator, which, for instance, prioritize rover safety and mission continuation over completion of particular tactical objectives. A set of plan robustness and scientific productivity metrics is generated as a result of these simulations, which can then be used for making plan improvements or for quantitative comparisons between different plans.

4.3 Landing Site Selection

Landing Site Selection, as mentioned previously, is a composite use case. It iterates over a list of candidate landing (mission start) times and invokes Mission Synthesis to generate an optimized reference mission plan for a candidate. Once the plan is generated, it is stress-tested by Mission Evaluation. When all the candidate sites have been processed, their evaluation statistics are compared according to user preferences and a ranked list of candidates is produced.

5. Application to the VIPER mission

SHERPA has been used extensively in the development of the VIPER mission, including for landing site selection, generation of optimized strategic mission plans for numerous candidate mission start dates, and system engineering studies. These uses are described in the rest of this section.

Copyright 2025 by the Canadian Space Agency (CSA) on behalf of SpaceOps. All rights reserved. One or more authors of this work are employees of the government of the United States of America, which may preclude the work from being subject to copyright in United States of America, in which event no copyright is asserted in that country.

5.1 Landing site selection

The VIPER landing site was selected from a set of 88 candidate sites. Each candidate site was defined as an area with the radius of 100 m, chosen to accessible by the lander from the planned final lunar orbit and satisfying lighting, communication, and terrain roughness requirements. Landing site candidates also had to be located at least 800 m away from the boundary of the nearest major PSR (to avoid contaminating the PSR with lander engines exhaust products, some of which contain hydrogen).

The landing sites were geographically grouped into six subregions, depicted in Figure 5. SHERPA was used to generate and compare mission plans originating out of each candidate subregion. By the process of elimination, Subregion 3 was selected as it provided the best options for visiting a scientifically important PSR located to its north-east and maximizing the scientific return and robustness of the overall mission. SHERPA was then used to down-select to the specific site by, once again, doing a comparison between the optimized traverses generated out of each landing site candidate within the subregion.

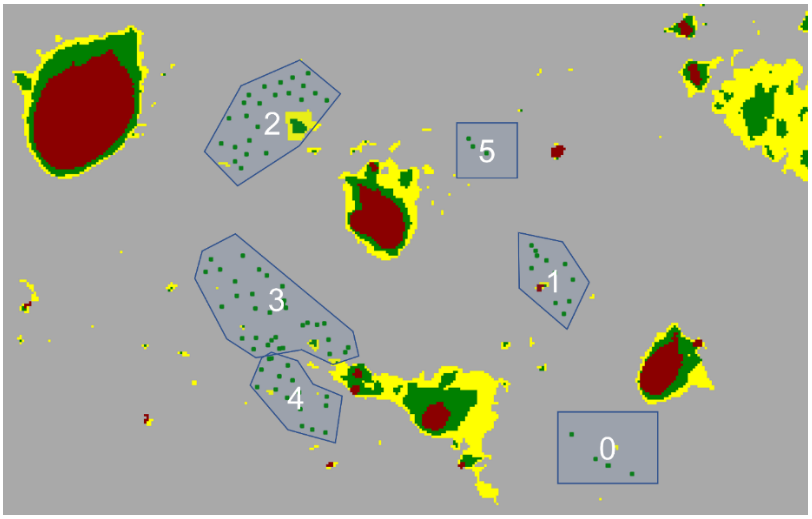

Figure 5. VIPER landing subregion candidates.

5.2 Mission plan generation

A comprehensive, multi-stage mission planning process was created for VIPER, with the overall planning pipeline depicted in Figure 6. Many of the data products used in the planning process are derived from the data obtained by the Lunar Reconnaissance Orbiter (LRO) [22]. Altimetry data images of the mission area produced by LRO are put through the Ames Stereo Pipeline (ASP) [23] to produce a digital terrain model (DTM). The DTM, in turn, is used to derive science and mission planning data products for the time period of interest. These include sun and communication

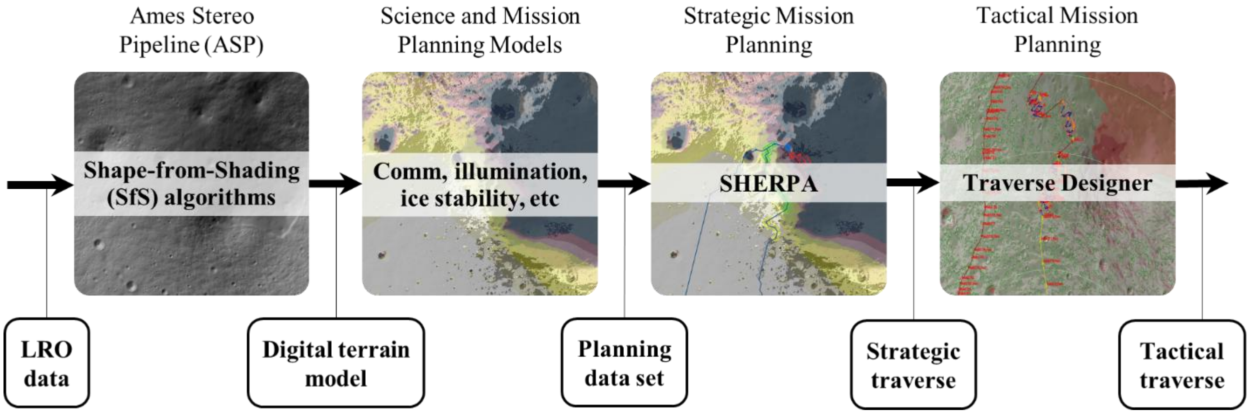

Figure 6. VIPER mission planning pipeline.

Copyright 2025 by the Canadian Space Agency (CSA) on behalf of SpaceOps. All rights reserved. One or more authors of this work are employees of the government of the United States of America, which may preclude the work from being subject to copyright in United States of America, in which event no copyright is asserted in that country.

shadow time series (typically generated with a 2-hour time step), time-to-shadow timeseries, slope maps, PSR boundary maps, safe haven maps (by lunar day), and ice stability depth models [24]. These products — together with science measurement plan, activity dictionary, vehicle models, sun and Earth azimuth/elevation above the horizon time series, and flight rules — form the planning data set.

The planning data set and user planning preferences serve as inputs to SHERPA's Mission Synthesis use case, The plan starts from the user-specified landing time and location and ends at the rover disposal site, spanning the entire duration of the mission. It includes both nominal and contingency branches, with locations and timing information for science stations, PSR entries, battery recharge activities, and safe havens. It also provides routing information between targets with 20x20 m/pixel resolution. An example strategic plan is presented in Figure 7.

The generated strategic mission plan can be interactively (and iteratively) modified by users through, for example, adding points of interest for the rover to drive through, changing branch-off conditions on contingency branches, or changing the type, location, and shape of a science station. After modifications (if any) are made, the plan is stress-tested through the Mission Evaluation use case. A set of plan robustness and scientific productivity metrics is generated as output, which can be used in making plan improvements or for quantitative comparisons between different plans.

Once finalized, the strategic mission plan is expanded into a tactical plan using VIPER's tactical planning tool (Traverse Designer). The details added in a tactical plan include a higher-resolution rover path, timing of communication passes, and activity assignments for individual mission control positions. It is anticipated that the strategic mission plan (and, therefore, its tactical counterpart) will be updated several times throughout the mission, particularly while the rover is parked at safe havens for up to two Earth weeks. The plans will be regenerated with the on-the-ground information obtained up to that point in the mission, including rover SMG, activity durations, and planning data products (e.g., DTM and ice stability depth models).

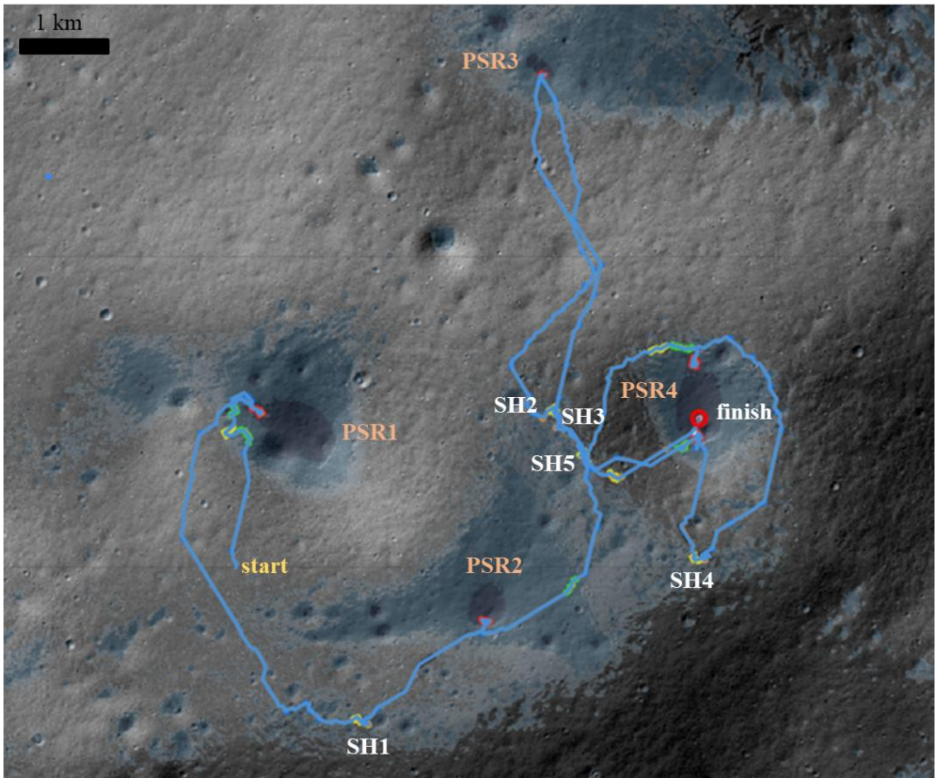

Figure 7. An example VIPER strategic traverse generated by SHERPA. 'PSR[number]' labels denote the permanently shadowed regions visited on the traverse and 'SH[number]' labels denote the safe havens for each lunar night.

5.3 System engineering studies

SHERPA was also used in two system engineering studies during VIPER development. The first examined the effects of reducing the number of solar panels on the rover from three to two (for mass savings) on scientific productivity and traverse robustness. The rover models were adjusted for the two-panel configuration and the

Copyright 2025 by the Canadian Space Agency (CSA) on behalf of SpaceOps. All rights reserved. One or more authors of this work are employees of the government of the United States of America, which may preclude the work from being subject to copyright in United States of America, in which event no copyright is asserted in that country.

performance statistics for the generated mission scenarios were compared to the statistics for the baseline, three-panel configuration. In the end, the three-panel configuration was retained.

The second study was related to the degradation of the rover's effective forward speed when driving in the sunward direction. Driving in the sunward direction (within approximately 45 degrees of the azimuth to the Sun) results in significant glare effects in the navigation and hazard avoidance camera images, thus necessitating driving in a zigzagging (tacking) manner, slowing down forward progress. SHERPA was used to generate mission plans with different sunward driving penalty scenarios to estimate the impact on the scientific productivity and robustness to disruptions, and, therefore, the need for mitigation measures.

6. Current work

SHERPA is being actively extended to new mission applications. It has been deployed to generate and evaluate traverses for the Pressurized Rover (part of the Artemis lunar program). While there are substantial differences in the mission parameters (e.g., longer traverses) and vehicle design (substantially higher speed and different recharging procedures, for instance), the transition to supporting Pressurized Rover was mostly limited to replacing the input data sets and modifying the MDP models. SHERPA is also being used to select the landing site and plan reference traverses for a Commercial Lunar Payload Services (CLPS) [25] mission.

A new SHERPA use case, being developed under NASA Ames called Robust Precomputed Autonomy (RPA) is being built to support bound-complexity applications with offline decision-making policies. These offline policies can be compiled down to, in essence, look-up tables, mapping system states to the optimal actions to take in these states to maximize the chances of attaining a goal state. Offline policies can be thoroughly verified, validated, and flight qualified. They also enable near-instantaneous response time and require minimal spacecraft power for computations.

In the initial phase of this work, we focused on developing methods for offline policy generation for individual spacecraft systems. We chose to construct the decision-making models as POMDPs to support reasoning under state uncertainty, especially in off-nominal situations (at the expense of significantly higher computational complexity). Enceladus Orbilander [26] was selected as the design reference mission, with the life detection instrumentation suite being the first Orbilander system modelled. As part of this work, we developed a methodology to encode certain types of domain knowledge as Bayes nets complementing a POMDP model. Finally, we validated the performance of the generated offline life detection model on test scenarios and demonstrated that it exceeds the baseline concept of operations policy both in accuracy and sample efficiency. Next steps in this work include developing a methodology for hierarchical reasoning, where a higher-level offline policy guides strategic action selection and manages the interactions between spacecraft systems.

7. Conclusions

The SHERPA decision support system has demonstrated that integrating probabilistic reasoning into mission planning and operations is not only possible, but also highly beneficial. During the development of the VIPER mission, it significantly reduced the workload of the mission planning team, allowing to generate and evaluate numerous non-deterministic mission scenarios in a small fraction of the time it would have taken to do so manually. It also provided the decision makers with quantitative metrics on the robustness of mission plans to likely disruptions and their expected productivity when execution deviates from the nominal plan. Additionally, it provides a real-time replanning capability for situations when attempting to adhere to the baseline plan during mission execution is no longer an option. SHERPA is currently being extended to support other missions in their development phase. A new module in SHERPA, Robust Precomputed Autonomy, is being developed to provide verifiable decision-making policies for missions with bounded state and action spaces, particularly those where a high degree of autonomy is required due to significant communication delays.

We hope that this work will not only pave the way to further acceptance of AI systems based on formal decision-making under uncertainty ground-based mission planning and control operations, but also lead to their eventual deployment onboard both robotic and crewed spacecraft. With that, we believe that SHERPA will contribute to making space missions safer, more affordable, and more productive.

Acknowledgements

The authors express deep appreciation for the support of this work by their colleagues on the VIPER mission, the Artemis program, and the CLPS program. The authors are also grateful for the funding for this work, provided by the NASA Ames Director's Discretionary Fund, the VIPER mission, the Artemis program, and the Ames Center Innovation Fund.

Copyright 2025 by the Canadian Space Agency (CSA) on behalf of SpaceOps. All rights reserved. One or more authors of this work are employees of the government of the United States of America, which may preclude the work from being subject to copyright in United States of America, in which event no copyright is asserted in that country.

References

[1] D. Gaines, R. Anderson, G. Doran, W. Huffman, H. Justice, R. Mackey, G. Rabideau, A. Vasavada, V. Verma, T. Estlin, L. Fesq, M. Ingham, M. Maimone and I. Nesnas, "Productivity Challenges for Mars Rover Operations," in *International Conference on Automated Planning and Scheduling, Planning and Robotics Workshop*, 2016.

[2] A. Colaprete, R. C. Elphic, M. H. Shirley, K. Ennico-Smith, D. S. S. Lim and K. Zacny, "The Volatiles Investigating Polar Exploration Rover (VIPER) Mission," 2021.

[3] N. Muscettola, P. Nayak, B. Pell and B. C. Williams, "Remote Agent: to boldly go where no AI system has gone before," *Artificial Intelligence,* vol. 103, no. 1-2, pp. 5-47, 1998.

[4] C. M. Meyer, C. Fulton, W. Maul, A. Chicatelli, H. Cannon, A. Bajwa, E. Balaban and E. Wong, "Propulsion IVHM Technology Experiment Overview," *IEEE Aerospace Conference,* 2003.

[5] K. Nicewarner and G. Dorais, "Designing and Validating an Adjustably-Autonomous Free-Flying Intraspacecraft Robot," in *AIAA Space Conference*, 2006.

[6] S. Narasimhan and L. Brownston, "HyDE -- A General Framework for Stochastic and Hybrid Model-based Diagnosis," in *International Workshop on the Principles of Diagnosis*, 2007.

[7] E. Balaban, H. Cannon, S. Narasimhan and L. Brownston, "Model-Based Fault Detection and Diagnosis System for NASA Mars Subsurface Drill Prototype," in *IEEE Aerospace Conference*, 2007.

[8] E. Balaban and J. J. Alonso, "A Modeling Framework for Prognostic Decision Making and its Application to UAV Mission Planning," in *Annual Conference of the Prognostics and Health Management Society*, 2013.

[9] E. Balaban, T. Arnon, M. H. Shirley, S. F. Brisson and A. Gao, "A System Health Aware POMDP Framework for Planetary Rover Traverse Evaluation and Refinement," in *AIAA Space Conference and Exposition*, 2018.

[10] D. R. Andrews, A. Colaprete, J. Quinn, D. Chavers and M. Picard, "Introducing the Resource Prospector Mission," in *AIAA Space Conference and Exposition*, 2014.

[11] J. Frank, A. Jónsson and P. H. Morris, "On Reformulating Planning as Dynamic Constraint Satisfaction," in *International Symposium on Abstraction, Reformulation, and Approximation*, 2000.

[12] J. L. Bresina and P. H. Morris, "Mixed-Initiative Planning in Space Mission Operations," *AI Magazine,* vol. 28, no. 2, pp. 75-88, 2007.

[13] J. L. Bresina and P. H. Morris, "Mission Operations Planning: Beyond MAPGEN," in *IEEE International Conference on Space Mission Challenges for Information Technology*, 2006.

[14] J. L. Bresina, P. H. Morris, M. C. Deans, T. E. Cohen and D. S. Lees, "Traverse Planning with Temporal-Spatial Constraints," in *International Conference on Automated Planning and Scheduling*, 2017.

[15] V. Verma, M. Maimone, D. Gaines, R. Francis, T. Estlin, S. Kuhn, G. Rabideau, S. Chien, M. McHenry, E. Graser and A. Rankin, "Autonomous robotics is driving Perseverance rover's progress on Mars," *Science Robotics,* vol. 80, no. 8, 2023.

[16] L. P. Kaelbling, M. L. Littman and A. R. Cassandra, "Planning and Acting in Partially Observable Stochastic Domains," *Artificial Intelligence,* vol. 101, no. 1--2, pp. 99-134, 1998.

[17] J. Kemeny and J. Snell, Finite Markov Chains, Springer, 1983.

[18] P. C. Fishburn, Utility Theory for Decision Making, Wiley, 1970.

[19] J. Bezanson, A. Edelman, S. Karpinski and V. B. Shah, "Julia: A Fresh Approach to Numerical Computing," *SIAM review,* vol. 59, no. 1, pp. 65-98, 2017.

[20] M. Egorov, Z. N. Sunberg, E. Balaban, T. A. Wheeler, J. K. Gupta and M. J. Kochenderfer, "POMDPs.jl : A Framework for Sequential Decision Making under Uncertainty," *Journal of Machine Learning Research,* vol. 18, no. 26, pp. 1-5, 2017.

[21] G. Chaslot, S. Bakkes, I. Szitai and P. Spronck, "Monte-Carlo Tree Search: A New Framework for Game AI," in *Artificial Intelligence and Interactive Digital Entertainment Conference*, 2008.

[22] R. Vondrak, J. Keller, G. Chin and J. Garvin, "Lunar Reconnaissance Orbiter (LRO): Observations for lunar exploration and science," *Space Science Reviews,* vol. 150.

Copyright 2025 by the Canadian Space Agency (CSA) on behalf of SpaceOps. All rights reserved. One or more authors of this work are employees of the government of the United States of America, which may preclude the work from being subject to copyright in United States of America, in which event no copyright is asserted in that country.

[23] O. Alexandrov and R. A. Beyer, “Multiview Shape-From-Shading for Planetary Images,” *ESS,* vol. 5, no. 10, 2018.

[24] M. Siegler, J. Martinez-Camacho, D. Paige, J. Williams, M. Shirley, R. Beyer, M. Hirabayashi, E. Costello, R. Elphic and A. Colaprete, “High Resolution Models of Polar Ice Stability,” in *Lunar Polar Volatiles Conference,* 2022.

[25] S. Cahill and D. Mauro, “Commercial Lunar Payloads Services (CLPS),” NASA Technical Report, 2022.

[26] S. MacKenzie, M. Neveu, A. Davila, J. Lunine, K. Craft, M. Cable, C. Phillips-Lander, J. Hofgartner, J. Eigenbrode, J. Waite and C. Glein, “The Enceladus Orbilander mission concept: Balancing return and resources in the search for life.,” *The Planetary Science Journal,* vol. 2, no. 2, 2021.