Journal of Aerospace Information Systems
Vol. 21, No. 1, January 2024



# Demonstrating Autonomy for Complex Space Missions: A Europa Lander Mission Autonomy Prototype

Caleb Wagner,* Cecilia Mauceri,† Philip Twu,‡ Yuliya Marchetti,§ Joseph Russino,¶ Dustin Aguilar,** Gregg Rabideau,†† Scott Tepsuporn,‡‡ Steve Chien,§§ and Glenn Reeves¶¶
*Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California 91109*

https://doi.org/10.2514/1.I011294

**There is a desire for robotic spacecraft to perform exploration in unknown, dynamic environments. The Europa Lander Mission Concept is one such mission that needs to deal with an extremely limited lifetime and energy supply, manage intermittent communications with long blackouts, face numerous environmental dangers, and ultimately take place too far from Earth to rely on human control. No missions to date have operated with the required level of autonomy and under the same level of communication constraints, uncertainty, and mission concept complexity as this mission. As a result, the viability of the autonomy must be demonstrated before it will be trusted with mission-critical planning. In this paper, we present an autonomous software prototype that can demonstrate and test the ability of different planners and executives to carry out complex, science-centric missions with limited interventions from humans. The prototype uses a hierarchical utility model that is used to maximize both the amount of expected science return as well as the number of mission objectives imposed by the ground. We demonstrate how this system handles some of the autonomous tasks expected of complex space missions such as decision making, in-situ data acquisition and analysis, data prioritization, resource management, and failure response handling in both simulation and on actual hardware. Through several scenario-based experiments we show how different planners and executives can meet the challenges of the Europa Lander Mission Concept. We also demonstrate that this system can be used in concert with a hardware prototype for autonomy field tests.**



## I. Introduction

As ROBOTS are tasked with exploring complex and unstructured environments under limited observability, they will need the means to act with little to no intervention from humans. This requires various autonomous behaviors, such as monitoring resource usage, making decisions based on sensor and instrument data, executing complex tasks in the real world, and determining what information should be communicated. The need for increased autonomy for robotic space exploration has been recognized in numerous reports, including the 2013–2022 Planetary Science Decadal Survey [1], the 2023–2032 Planetary Science Decadal Survey [2], the NASA Astrobiology Strategy [3], and the National Academy of Sciences Astrobiology Strategy [4] reports. An example of a mission that requires a larger degree of autonomy than any space mission to date is the Europa Lander Mission Concept [5,6]. The Europa Lander Mission Concept would send a robotic lander, depicted in Fig. 1, to Europa, an icy moon of Jupiter, and look for signs of life. The mission plan includes excavating at least 10 cm below the icy surface, collecting and analyzing material samples for potential biosignatures, and communicating the data back to Earth. While the science mission may sound comparable to prior missions, Europa poses additional challenges. No surface mission has been performed on Europa, and most knowledge about Europa comes from satellites orbiting Jupiter [7]. Therefore there is extremely limited information about Europa at this point in time, and even less information about the surface and subsurface characteristics. While the Europa Clipper mission [8] is expected to arrive at Europa in 2030 before a potential Europa Lander mission, the Clipper mission will likely not provide the detailed surface characteristics required to plan a surface mission. Communication with the Lander will also be limited. The distance to Europa results in a 45-min communication delay. Additionally, every 42 out of 84 h there will be a communication blackout with the Earth due to Europa's orbit. Finally, the Lander's mission is planned to span only 30 days; this is due to both a nonrechargeable battery energy source and the extremely inhospitable environment primarily caused by radiation from Jupiter. The short lifetime of the Lander means that efficient use of mission time is imperative.

The limited lifetime, extreme uncertainty, and communication blackouts require the Lander to act as quickly as possible. Specifically, the mission requires an autonomous system that can make planning decisions on its own with limited human intervention. This includes the following design objectives:

1) *Objective 1: System must maximize the number of completed mission objectives*. The Lander must attempt to accomplish as many of its objectives specified by the ground as possible.

2) *Objective 2: System must maximize the expected overall science return*. The Lander must attempt to accomplish its objectives in such a way that produces the greatest amount of positive scientific results. Note that this is different from Objective 1 as mission objectives may not maximize the expected science return, such as if the Lander must wait for approval from the ground before continuing. The Lander is expected to obey all mission objectives such as waiting for the ground but do so in a way that maximizes expected science return.

Received 12 April 2023; revision received 10 July 2023; accepted for publication 13 July 2023; published online 8 September 2023. 

*Autonomy Prototype Deputy Team Lead, Member of Technical Staff, Artificial Intelligence Group, 4800 Oak Grove Drive; caleb.t.wagner@jpl.nasa.gov (Corresponding Author).

†Autonomy Prototype Development Lead, Mobility and Robotic Systems, 4800 Oak Grove Drive.

‡Autonomy Architect, Task Manager, Robot Operations and V&V, 4800 Oak Grove Drive

§Element Lead for Instruments and Analytics, Machine Learning Group, 4800 Oak Grove Drive.

¶MEXEC Team Lead, Senior Member of Technical Staff, Artificial Intelligence Group, 4800 Oak Grove Drive.

**TRACE Team Lead, Member of Technical Staff, Maritime and Multi-Agent Autonomy, 4800 Oak Grove Drive.

††Principal Member of Technical Staff, Artificial Intelligence Group, 4800 Oak Grove Drive.

‡‡Autonomy Prototype Development Lead, Mobility and Robotic Systems, 4800 Oak Grove Drive.

§§Element Lead for Scheduling and Execution, JPL Fellow, Senior Research Scientist, Artificial Intelligence Group, 4800 Oak Grove Drive.

¶¶System Level Autonomy Lead, JPL Fellow, Systems Engineering Group, 4800 Oak Grove Drive.

**Fig. 1 An artistic depiction of the Europa Lander is shown top left. Photographs of a Prototype Europa Lander, shown on the right, and a collected sample, shown bottom left, were taken during a field test in Alaska. The field test used our autonomy software to collect the sample, respond to off-nominal events, and communicate the data to a ground station. This test demonstrated the capabilities of our autonomous system in a real-world environment. (Images courtesy NASA/JPL-Caltech.)**

3) *Objective 3: System must determine what information to send home and when*. Limited bandwidth means that not all information onboard the Lander will make it back to Earth by the time the Lander runs out of energy. As such, the autonomy must prioritize what information to send to the ground. This includes information needed to convey the intent of the Lander, as ground operators want insight into what the Lander is planning to do as well as why it is choosing to perform (or not perform) certain actions.

4) *Objective 4: System must manage its finite resources*. For example, with its limited battery, the Lander must consider the energy impact of every action in order to have enough energy remaining to achieve its science objectives.

5) *Objective 5: System must react to off-nominal conditions and hardware degradation*. Due to the uncertainty about the environment, the Lander autonomy needs to have the flexibility to respond to unexpected events. It should act as a closed-loop system where it both recognizes off-nominal conditions and takes appropriate actions to remedy them.

6) *Objective 6: System must collect data from its instruments and interpret its own scientific findings to plan*. In-situ analysis of collected samples will need to be performed. The Lander will need to determine for itself if it has found a biosignature and how to proceed, acting as a closed-loop system with respect to science.

The contributions of our paper consist of implementing an autonomous prototype that can model the mission challenges and investigate how different planners and executives fulfill these objectives. Specifically, we use a hierarchical utility model that allows the system to compare different actions and select the most promising ones while respecting the science objectives of the mission (objectives 1, 2, and 3). High-level tasks (such as mission goals) and low-level tasks (such as behaviors needed for survival) can be reconciled in this architecture. Furthermore, the proposed framework allows for mission designers to communicate the relative priority of actions so that the system will accomplish tasks according to the intent of the designers (objectives 3 and 5). This priority can be made context dependent so that the priority of tasks will change throughout the mission. We also simulate Lander hardware, such as science instruments and energy systems, as well as the Europan environment, such as frigid temperatures and Europan quakes (objectives 4 and 6). Using the prototype, we can test how a planner or executive responds to different mission scenarios to achieve all the objectives listed above. The prototype can be integrated with physical hardware, such as to conduct a field test as described in Sec. V.D. Note that while we did consider the efficiency of all algorithms and designs, we did not restrict our design based on available computational power. A current subprogram of the Europa Lander Pre-Project worked on having the necessary hardware to meet our computational needs. While also not a strict requirement, we wanted a system that could reproduce the same results when run with the same inputs so that we could more easily verify the autonomy works as intended.

This paper is ordered as follows. First, a literature review is performed to compare our autonomy architecture to existing autonomous missions as well as existing autonomy architectures. Next, an overview of the system autonomy is described followed by a description of the various components that make up the full autonomy architecture. We address how our system handles various autonomous behaviors, including autonomous decision making, in-situ data acquisition and analysis, data prioritization, resource management, and failure response handling. We also describe how this software architecture acts as a simulation testbed in order to evaluate different autonomy algorithms. Finally, an analysis of the autonomy architecture is performed based on its ability to respond to various goals and challenges taken from the Europa Lander Mission Concept.

## II. Related Work

Comparing the Europa Lander Mission Concept to existing robotic surface missions highlights the need for increased autonomous capabilities. Mars serves as a good comparison to explain why heritage software cannot be directly used. Mars surface missions are generally planned on the order of years, and there is considerable history of Mars missions lasting for much longer than their nominal science mission. This is fundamentally different from the Europa Lander Mission Concept, which is expected to last for no more than 30 days, presenting the challenge of an extremely limited lifespan.

Despite the relatively large bandwidth when compared to Europa, the Mars surface operations still must deal with constrained communication and limited information, such as the limited availability of sun-synchronous orbiters that can result in delays in mission execution [9,10]. This poses another major challenge for the Europa Lander Mission; not only will the mission have constrained communication like Mars, but it will also have communication blackouts that occur for 42 out of every 84 h. These communication and information constraints restrict mission planners with how far in advance they can plan and how detailed those plans can be. Planners for Mars missions will create a variety of plans, such as strategic (30–90 sols in advance), supratactical (7–10 sols in advance), and tactical (1–3 sols in advance) plans [11]. The tactical plan defines the actual activities that will be executed on the rover; this plan cannot be generated too early in advance due to changing environmental conditions that will affect traversability and close-contact operations, as well as limited information such as the inability to select specific science targets until additional data are collected from the rover (as orbital imagery is usually not sufficient) [9,10].

Because of this, mission planners use the results of activities performed during the previous sol to make decisions about the current sol [9]. The decisions involve planning, control, and failure response and are carried out by teams of experts from the Payload Downlink Lead (PDL) to the Science Operations Working Group (SOWG) [12]. These mission planners author sequences (a set of specific actions) containing hundreds of commands and thousands of arguments that are uploaded to the robot; the robot carries out the sequences exactly as scripted by the ground [13]. Every new and changed sequence uploaded each sol requires command-by-command review by the team of experts [14]. This kind of ground-in-the-loop (GITL) decision making would be too time-consuming for the short duration of the Europa Lander Mission, especially given the desire not to have the Lander do nothing while waiting for the ground to make a decision during the approximately 42 h communication blackout between Earth and Europa.

For Mars missions, the ground team is also heavily involved in planning the scientific mission. The 2007 Phoenix mission used a robot arm to excavate and sample soil and ice [15,16]. Information about the surface was already known through the Mars Odyssey spacecraft [17] as well as the prior Viking lander mission [18]. With this prior information, a plan to perform excavation was generated offline rather than in real-time. Downlinked images were used by the ground team to generate command sequences used to control the lander [15]. This same method was used for the Insight mission, where the ground used downlinked panoramas and terrain meshes to select where to deploy instruments [19,20], as well as for current rover missions such as MSL, where downlinked terrain meshes are used by ground operators to select and upload target sites for sampling [14]. Newer missions such as the M2020 (Perseverance) mission have increased automated capabilities such as scheduling activities with preheats and autogenerating awake activities [21]. However, these plans are still uploaded from the ground, and the ground has full control over the order of activities. The Mars Ingenuity Helicopter also has several autonomous capabilities, such as taking off, flying, and landing autonomously; however, it still follows a set of predetermined, manually selected waypoints when flying [22] and must land and take off from “certified” safe airfield as it does not have automated safe landing site detection capabilities [23].

The short 20–30-Earth-day (5–9-Europan-sol) lifetime of the Europa Lander means that the science objectives must be completed as soon as possible to allow ample margins for exogenous events and contingencies. The nominal surface operation timeline, described by [24], shows that an area of the surface must be selected and an excavation performed on the very first sol after landing. With this extremely short turnaround, only one GITL cycle can be performed before the excavation site is both selected and excavated. Previous missions, such as Phoenix, have used multiple GITL cycles to assess different candidates [25]. As such, in order to meet these objectives, the decision-making process must be greatly expedited. One of the best ways to achieve this is by allowing the Lander to make its own science assessments and activity selection. The ground team should validate these decisions rather than make them.

Autonomous decision making has been performed in existing missions and is expected to be used in future missions. Most practical uses of autonomy, however, has not been performed at the complexity and scope required by the Europa Lander. For example, Autonomous Exploration for Gathering Increased Science (AEGIS) is a system used by the MSL rover to collect observations of autonomously selected scientifically interesting targets [26,27]. While AEGIS is used to identify science targets and take observations, the data are processed by the ground, and the ground makes planning decisions based on the collected data. As described by [14], many different automated planning and execution approaches have been built such as for Earth orbiters [28,29] and prototype deep space missions like Europa Clipper [30], but none have been used to date in surface missions. Gaines et al. [9] describe the productivity that autonomy might afford existing Mars missions. The paper details how managing operations on Earth do not always align with activities on Mars such as when data can be downlinked and uplinked. The scheduling mismatch causes periods where little to nothing is performed as the team waits until the next Mars morning to uplink products to the rover. Suggestions to improve productivity include performing autonomous in-situ science as well as using an onboard planner [9,21,31,32]. These features are included in our autonomy prototype. Recent work such as the Mars 2020 onboard planner [33] is capable of in-situ science and planning, but the design is highly tailored to the mission and only focuses on variable activity durations while the Europa Lander Mission Concept requires additional capabilities such as reprioritization of activities and the abilities to respond to exogenous events.

Our autonomy prototype architecture most closely resembles Framework for Robust Execution and Scheduling of Commands On-Board (FRESCO) and Coupled Layer Architecture for Robotic Autonomy (CLARAty). FRESCO is a framework that relies on state-based goal definitions and centralized management of state knowledge. It employs hierarchical reasoning to provide clear abstractions between different resources [34]. Inspired by FRESCO, the autonomy prototype also includes these features. FRESCO does not include data prioritization or in-situ analysis, key elements of our system. CLARAty [35] is also an inspiration for our autonomy architecture with many concepts used in its design such as separating the architecture into a planning layer and a functional layer. We expand on the capabilities of decision making and goal monitoring described in the CLARAty paper through our hierarchical utility model.

There are many other existing autonomy frameworks that integrate or have the potential to integrate science autonomy with robotic autonomy, such as Robot Operating System Plan (ROSPlan) [36], Europa [37], “C” Language Production System (CLIPS) [38], Plan Execution Interchange Language (PLEXIL) [39], and Reactive Model-based Programming Language (RMPL) [40]. The Remote Agent (RAX) [41] is one example of an existing autonomy framework that has flown in space and controlled the Deep Space 1 spacecraft for two periods of time totaling about 48 h. These architectures all work on the same premise of updating state information using sensor reading; a planner is able to query the current world state and generate a plan. In addition, they employ an executive to oversee the execution of the plan and adapt the plan to off-nominal events. These frameworks sometimes struggle with specifying relative priorities of tasks.

Missions like the Europa Lander can be thought of as multi-objective decision-making problems, where there are several competing objectives, including maximizing science return, maximizing the number of mission objectives that are achieved, as well as ensuring that resource constraints are met. There are several ways that these competing criteria might be implemented through existing autonomy architectures. One way is to impose constraints on the ordering of actions to meet the mission objectives like those used by ROSPlan [36]. While this strategy can be effective, it may produce suboptimal plans by imposing unnecessary time constraints. Another way to balance science return and mission objectives is scaling utility to achieve some preference-based ordering as described by [42]. The greatest challenge in using this strategy comes from lost contextual information. For example, a plan that has a very high science return but meets only a handful of the science objectives (i.e., a single collected sample with a positive biosignature) may be treated equal to a plan that achieves relatively low science return but meets a large number of the science objectives (i.e., two collected samples with negative biosignatures as well as seismographic and panorama data). Implementing this strategy successfully requires very careful consideration of the weighting of the science objectives so that the combination of different actions does not have unintended consequences. This is infeasible for a mission like the Europa Lander, which has a very large set of behaviors and objectives. A third strategy, as described by [43], is to balance the two maximization criteria of science return and number of achieved mission objectives by calculating a set of policies such that for every possible combination of the maximization criteria, a maximizing policy is in the set. Generating multiple plans in this way requires computational processing that is not available for space missions like the Europa Lander.

Our utility model avoids the problem of choosing appropriate weights by using a hierarchical data structure that preserves both the achieved mission objectives (design objective 1) as well as the science return (design objective 2). The concept of a hierarchical utility model is not new. Wellman and Doyle [44] use utility trees as a tool to weight maximization criteria. The utility trees generate a scalar utility value that could be used to compare which action to take next. Rather than collapsing to a scalar value, our hierarchical model preserves strict preferences between the different tasks. A major constraint of this prototype is that the Lander is expected to follow the rules and guidelines imposed on it by mission operations regardless of the likelihood of preferable results. For example, if it is possible to collect a sample, even if the likelihood of that sample producing a positive biosignature is extremely small, the action is still preferred over collecting more seismometer data, which might have a higher probability of producing high-value science information. As such, our model is not considering *expected* utility, but instead only utility. This can be seen as an extension built upon the work in [42]. Our model forces the planners to adhere to the mission objectives; even if the Lander could collect an infinite number of seismometer data, if the mission planners place a strict preference on collecting a sample, the Lander should pick the plan that collects one sample over the plan that has an infinite amount of seismometer data but no collected sample. This concept of selecting one action over another regardless of the actual utility or expected utility value is referred to as dominance and is further described in Sec. III.A. The strategy of dominance between actions is used to guarantee that the Lander will (attempt to) accomplish certain objectives, whereas a classic utility model makes no such guarantees. If the units for utility are all the same as is commonly used in classic utility theory, it is possible for the planner to prefer selecting several lower-reward goals rather than a single higher-reward goal.

This approach still affords the planner flexibility in selecting actions when the mission objectives are underspecified or partially ordered. For example, if collecting seismometer data and collecting episodic panoramas are preferred equally (in other words they are at the same level in the hierarchical model), the Lander might choose to take 10 panoramas and no seismometer data, 10 seismometer data and no panoramas, or some combination of the two. Although this is beyond the scope of this paper, it is possible to incorporate expected utility here by considering the likelihood of receiving a given utility for every task in the same level (but not with respect to the likelihood between levels). Basich et al. [45] as well as an upcoming related paper more focused on the planners mentioned in this paper describe other approaches for dealing with uncertainty during planning. This includes using periodic replanning and sampling-based optimization to handle uncertainties that may arise.

Unlike [43], our hierarchical model allows mission designers to reason about the utility of tasks at various levels of abstraction and implement new rules without having to consider the numerical utility value assigned to a task. This also allows for a simplistic comparison of different plans as it is immediately clear which objectives or criteria caused one plan to be preferred over another as further described in Sec. III.A rather than just a numeric ranking of plans. Existing research has also looked at the benefits of hierarchical models in planning domains for hierarchical task networks. Lekavý and Návrat [46] compare hierarchical task nets (HTNs) to STRIPS planning domains, which are similar to scalar utility models, and show that they have similar expressivity. Erol et al. [47] also point out that HTNs are a user-friendly way of injecting domain knowledge into the task net, which is similar to the concept of dominance in our hierarchical utility models. A major element of our autonomy is that it must work with human experts; this requires building a system that the experts can understand, which can be achieved by building a good mental model of the planner.

Note that we do not make any claims regarding the optimal science return of the Lander using our model. It is entirely possible that an expected utility model that does not support our concept of dominance is able to perform additional science as the expected utility model does not have to spend time performing actions for which the likelihood of a high-value science return is low. As described earlier, heritage NASA missions such as Mars missions involve sophisticated missions operations teams that perform a lot of the high-level planning for the given robot. This strategy has proven highly successful. Our approach for the Europa Lander Mission Concept is to strike a balance between the two sides of manual control and full autonomy. The Lander is expected to follow the high-level orders of the mission operations (design objective 1) but is able to autonomously determine the actual execution of said orders to maximize science return (design objective 2).

The work described in this paper was part of the larger Europa Lander Advanced Development project. This project supported several research threads crucial to advancing the technological readiness for the Europa Lander Mission Concept. Among the related projects, the work on sampling autonomy, the Blackbird simulator, and the ground operations design simulations fed directly into this autonomy prototype design. Sampling autonomy is the control and planning system used to generate trajectories for the Lander's robotic arm [48]. This is a separate autonomous system from our prototype. The prototype can command the sampling autonomy system to perform actions but otherwise treats it as a black box. We demonstrate integration with the sampling autonomy system in Sec. V.B. Blackbird [49] is a tool that has been developed using a modeling and simulation framework to quickly evaluate mission and operational concepts. It is not a real-time simulation like the autonomy prototype. The work performed by Blackbird and through other mission modeling efforts, such as the work done by [50], has been used to drive the mission objectives and feature requests for the prototype. Finally, as part of Europa Lander Advanced Development, several design simulations for ground operations were performed. Despite being an autonomous system, human operators will still be involved in the mission, monitoring the progress of the Lander and overriding decisions when necessary. As such, it is critical to understand the best ways to communicate the state of the Lander and provide sufficient situational awareness for the human operators such that the human operators and autonomy can work together. For each design simulation, large teams of scientists and engineers were brought together to experience a realistic operations scenario of different aspects of a Europan mission. A paper describing in much greater detail the methodology and findings of the design simulations is currently under development. These design simulations were crucial for providing early and realistic feedback into the autonomy prototype and ground operations software by evaluating the thought processes and desired features of the mission's ops teams. During the simulations, data were provided using the instrument simulations from our software, and constraints of the autonomy prototype were enforced. While the design simulation did not directly use our software, they did validate many of the existing behaviors and the overall autonomy architecture for the Europa Lander.

Other work has developed both hardware and software testbeds to enable researchers to develop technologies relevant to the Europa Lander Mission Concept and other mission concepts. The Ocean World Lander Autonomy Testbed [51] is a physical testbed that simulates a lander with a robotic arm and sampling tool in low gravity. Edwards et al. [52] describe the OceanWATERS software simulation testbed that provides a high-fidelity simulation of many of the physical properties of Europa, such as the expected surface geometry and terrain dynamics as well as a 3D model of the Lander. Our architecture can be used to replace the generic autonomy module in OceanWATERS. The paper of Touma et al. [53] is another paper that has looked at autonomy for the Europa Lander, specifically recognizing and handling failures or anomalies during the mission. This work is parallel to ours as the architecture focuses on using the MONSID diagnostic tool to estimate the health of the Lander and communicate this information to the planner. As described in the paper, this tool can be integrated with TRACE, which is one of the executives we tested our system with as described in Sec. V.A.2. McMahon et al. [54] and its predecessor [55] propose the development of an autonomy architecture that, like ours, is used to maximize science return. The papers describe a Shared Science Value Map (SSVM) that is used by their task planner to determine the location surrounding the Lander that should be sampled. This is similar to our hierarchical utility model, except that the authors do not explain how their model can be generalized to other (lower-priority) tasks such as collecting seismometer data and taking panoramic images. Furthermore, their simulation assumes that the science value of a given task or data product is already known (left as future work), while our model calculates this information onboard.

In contrast to the aforementioned work, the work described in this paper was part of the Europa Lander Pre-Project, and therefore directly embodies the best available knowledge for a potential future Europa Lander Mission. As such, the autonomy software described in this paper was integrated with several flight-like flight software components that were tested with actual hardware in multiple venues, culminating in field trials in Alaska as described in Sec. V.B.

## III. Europa Lander Autonomy Prototype

The Europa Lander Autonomy Prototype is a simulation of the environment, hardware, and software of a Europa Lander Mission. We developed this prototype to serve as an autonomy testbed, from low-level autonomy, such as autonomic heating and energy management, all the way up to high-level autonomy, such as task planning and execution. As such, it must meet the mission objectives described in the Introduction, namely, making decisions with limited human intervention, managing finite resources, reacting to off-nominal conditions, collecting data and interpreting findings to (re)plan, and determining what data to send home and when.

To model these complex objectives, the Europa Lander system architecture, as shown in Fig. 2, is composed of several different modules. The planning and execution and system behavior modules shown in the diagram are responsible for choosing tasks and selecting when to execute them (objective 1 and 2). These modules also monitor tasks for off-nominal behaviors (objective 5). The energy and thermal autonomics modules are responsible for resource management by estimating the amount of remaining energy and current temperature of the system (objective 4). The instrument and analysis modules, along with the sampling subsystem, collect information about the surrounding environment, produce data products, and analyze data products (objective 6). The data product manager, downlink manager, and communications manager all work together to determine what data to send home and when (objective 3). Finally, the System State Manager is a database where the various modules share data to enable autonomous decision making. For example, the planning layer must be informed of the latest scientific findings of the instruments in order to adapt its behaviors to perform the greatest amount of science.

In this section, we will describe each of these modules, how they contribute to meeting mission objectives, and how they support autonomy validation.

### A. Autonomous Decision Making

The planning and execution layer is used to make decisions and then robustly execute those decisions. Due to the modularity of the architecture, the autonomy is agnostic to the actual planner or executive that is used so long as it can support the hierarchical utility model. This layer is used to achieve the design objective of maximizing the number of completed mission objectives (objective 1), maximizing the expected overall science return (objective 2), managing finite resources (objective 4), as well as reacting to off-nominal events (objective 5).

The hierarchical utility model maps mission-level behaviors to scientific measurements by grouping raw sensor data into abstract data structures that represent the task-level value of the scientific findings. The hierarchical model proposed in this paper ensures that the priority of tasks as determined by the mission designers is preserved. In other words, the mission objectives are maximized directly through the hierarchical structure of the model; the priorities of the objectives are compared such that higher-priority objectives dominate (are preferred over) lower-priority objectives. Note that lower-priority objectives can still be achieved, but a plan that has a single higher-priority objective will dominate a plan that contains several lower-priority objectives. At the same time, combining multiple science values at different levels within the hierarchy ensures that the science return can be maximized. The utility model also provides greater explainability for the mission designers as the sequence of information used to make decisions is encapsulated in the utility model.

There are two different types of utility described in this paper, task utility and plan utility. Task utility is the utility assigned to particular actions, while plan utility is the summation of all task utilities for a given set of actions. We will first describe the hierarchical model for task utility and then describe how to select a plan.

#### 1. Task Utility Calculation

We represent the task utility model as a tree, with the leaf nodes of the tree representing scalar utility values that are organized hierarchically according to mission priorities. The task utility hierarchy is shown in Fig. 3 with increasing levels of abstraction from left to right. At the left-hand leaf nodes, values are assigned to sensor measurements based on their information content, or science value. Measurements that contain "interesting" data such as recording a Europan quake will have a higher science value. At the next level, the science values are combined to score data collection tasks. These values allow for easy categorization and comparison of the science objectives that a plan or task achieves. This layer is used to quickly disregard plans that do not contribute toward the science objectives of the mission. In addition to science objectives, the top level of the model represents mission-level constraints that may be imposed on

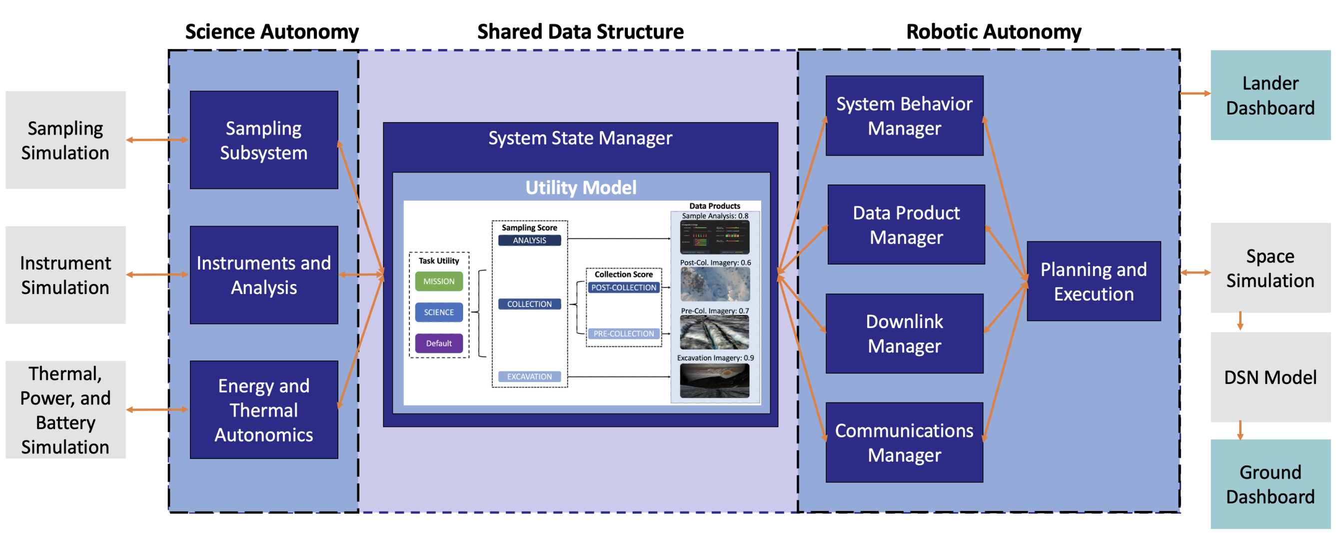



**Fig. 2 Overview of the entire autonomy architecture where both the science autonomy components and robotic autonomy components interact through the shared hierarchical utility data structure in the center.**

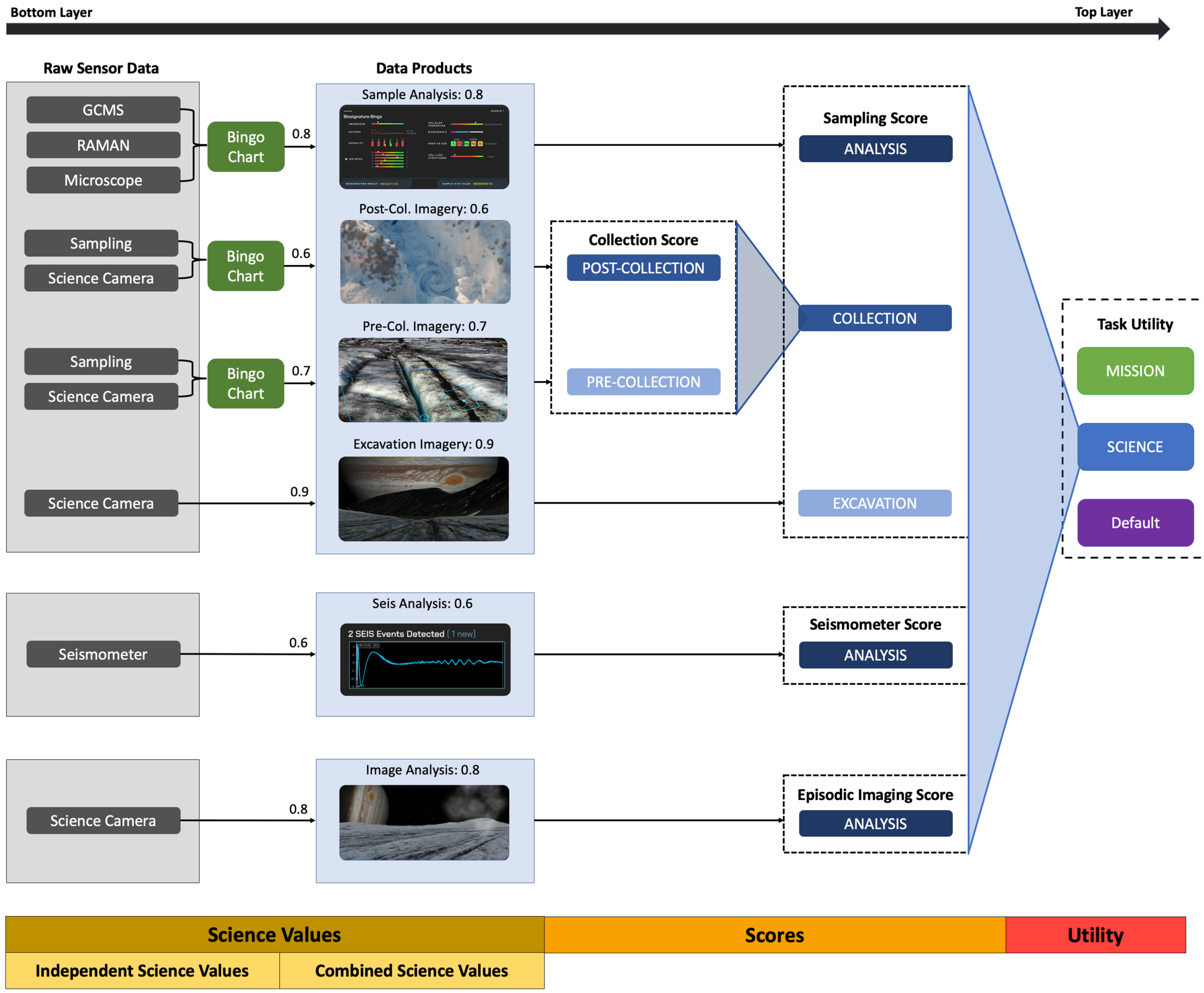



**Fig. 3 Overview of utility model hierarchy showing how data products shown on the left are grouped into scores that are grouped into utilities. The numeric value of each level is determined by the science value of the data product, which itself is determined by combining the outputs of various sensor data. (Data Products graphics created by Krys Blackwood, JPL.)**

plans, such as limiting the number of samples that can be collected from a given location.

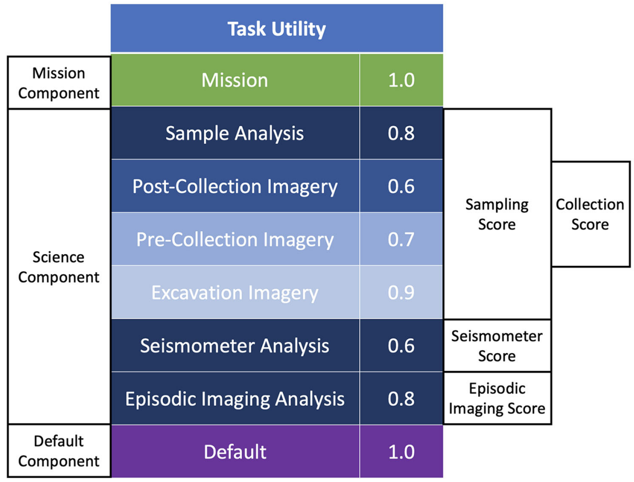
By performing a tree traversal, we can compare the utility values in hierarchical order. In practice, we flatten the utility model to an array, as shown in Fig. 4a. The flattened model represents the traversal order while allowing for more rapid comparison of utility values. The labels of the diagram also show how values can belong to multiple hierarchies, such as the precollection imagery (0.7), which belongs to the collection score, the sampling score, and the science component of the given task utility. Note that precollection imagery is an image taken before the Lander has collected a sample, while postcollection imagery is an image taken of a sample after it has been collected.

The mission component has the highest priority as represented by being first in the tree traversal. The mission component is a leaf node that represents the objectives the ground team wishes to enforce. For example, the mission component is used to enforce the rules of site selection so that no more than three samples are collected at a given site. Another mission rule might be that episodic panoramas must be taken at least 15 min apart from any previous panoramas. As such, if the planner considers adding a task that violates this rule, the mission component will have a very low value. While any value between [0,1] can be used for this utility, we generally choose either 0 or 1 as most rules represent binary constraints.

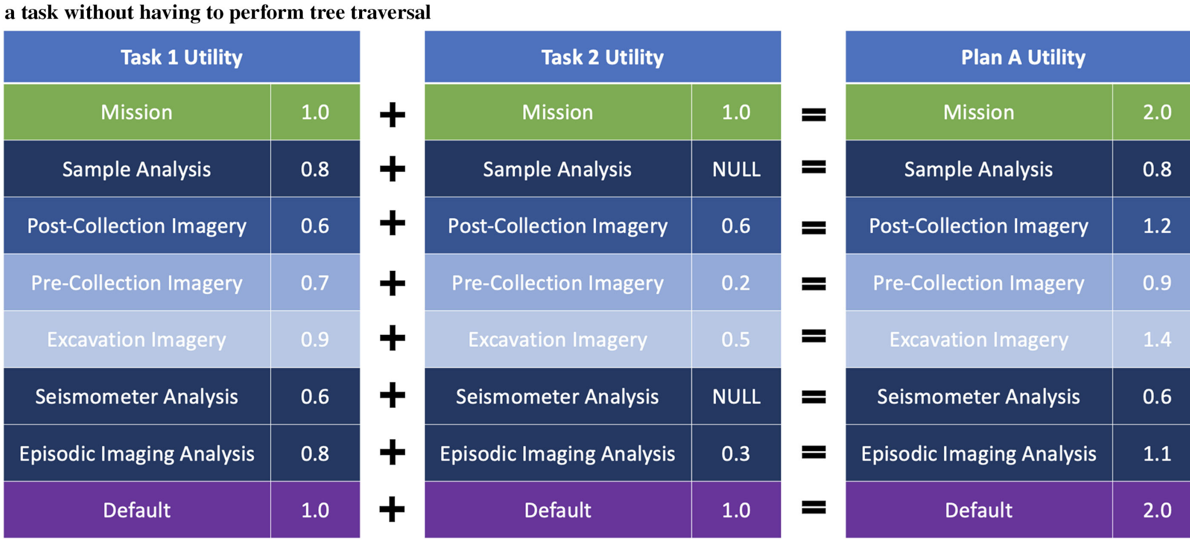
The next node in the top layer is the science component. The science component is a subtree that combines scores from all the science objectives that are achieved by a given task. Scores may be leaf nodes such as the analysis sampling score, or may branch to other scores such as the collection score. These additional branches can be thought of as subobjectives that are achieved by a given task.

The calculation of utility at leaf nodes can be different for each leaf. For example, for the science component, the leaf nodes represent data products from different science instruments. Each data product is analyzed to produce a science value (see Sec. III.B). A science value is a numeric value between 0 and 1, which reflects the scientific importance of the data. For example, if analysis of a collected sample indicates a positive biosignature, the science value would be high, indicating that the data product is of extreme interest to the ground. Conversely, the lack of a biosignature in a data product should cause the science value to be lower. Note that a leaf will only be assigned a science value if the data product is scheduled to be downlinked; otherwise it is assigned a value of 0. Because the Lander is expected to not just perform science but also communicate that science with the ground, science values are assigned only when data products are destined to be communicated with the ground.

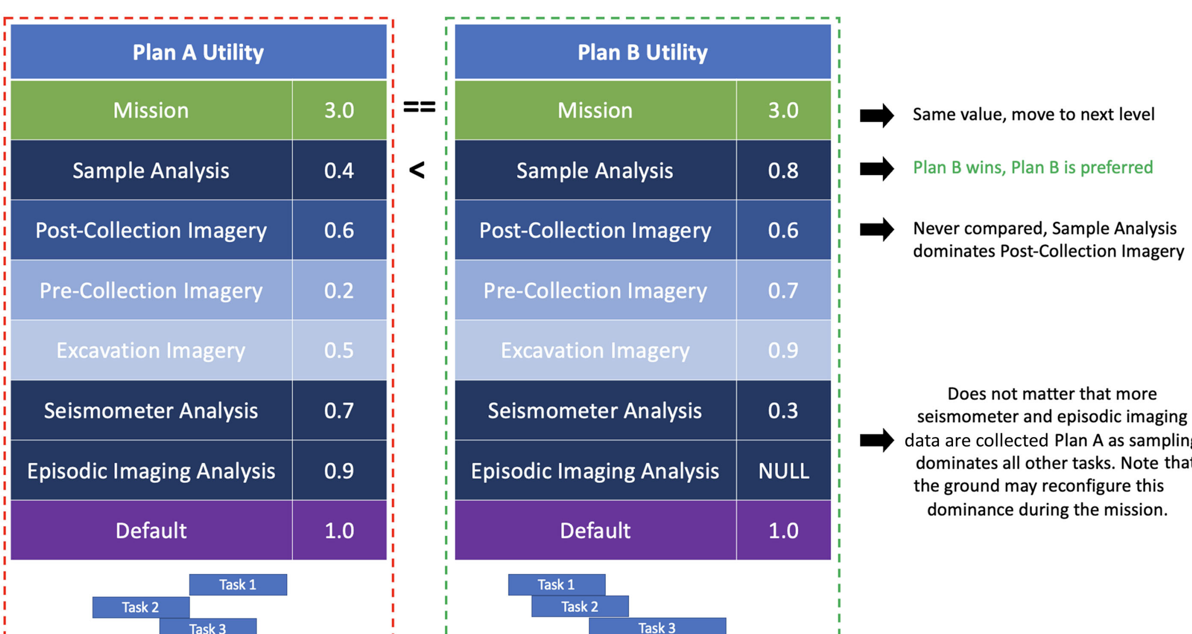
The sampling score component groups several data products together that represent the achievement of the given science objective. Because the scores map data products to science objectives, they can be used to compare different parameters for the same task. For example, in the Europa Lander, it is necessary to compare excavation outcomes for the different sites from which the Lander may choose to




**a) A flattened view of the hierarchical utility model shown in figure 3 that allows for rapid comparison of different utilities within a task without having to perform tree traversal**




**b) Plan utility is calculated by summing the components at the same hierarchy for two or more task utilities**




**c) Plan utilities are compared in a top to bottom fashion where the plan with the highest value at the highest level in the model is considered better**

**Fig. 4 A visualization of the calculation of task and plan utility using our hierarchical model.**

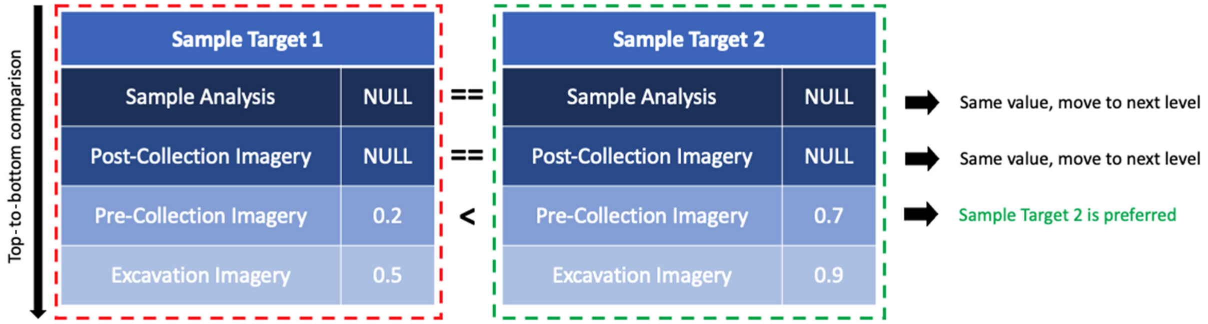



**Fig. 5 Sample Target 2 is preferred over Target 1 because of its higher precollection imagery value. This shows how the planner can adapt behaviors to the expected science return even when not much information is known about a given target.**

collect a sample. By following tree traversal order, the comparison still works when there is limited information about the sites, such as in Fig. 5. In this example, Sample Target 1 is preferred over Sample Target 2 even without access to sample analysis or postcollection imagery. The traversal continues until postcollection imagery is available for comparison.

The default component is the last (leaf) node in the first level of the hierarchy. The default component is used to assign a numeric value to each task. This value is important for tasks that do not achieve any specified science objectives but are still important to the mission, such as preparing the high-gain antenna to transmit data products. Practically, the value is almost always set to one so that it is better for a plan to include the task than to not include the task.

### 2. *Plan Utility Calculation*

Plan utility is the utility of performing a set of tasks, or plan. Comparing plan utilities is how a planner chooses between alternate plans and partial plans. To calculate the utility of a given plan, utility trees of the tasks are summed as shown in Fig. 4b. In the event that there is no matching component (i.e., Task 2 does not have a sample analysis), its value is assumed to be zero during the summation. Once the plan utilities have been calculated by summing the task utilities, the plan utilities can then be compared as shown in Fig. 4c. This is done by comparing each component in tree traversal order. If any one node has a higher utility, the traversal is stopped and the tree with the higher node wins. In the event of a tie, the next highest ranked components are compared. For example, in Fig. 4c, the mission components have the same score. Therefore, the comparison continues to the sample analysis node, where Plan B is preferred over Plan A because Plan B's sample analysis utility is higher. Had the sample analysis components been the same, the comparison would have continued to the postcollection imagery components, and so on.

In comparing the components of the utility model in this way, the hierarchical model ensures total dominance of certain activities. For example, sampling activities are considered more important than seismometer activities. This is reflected in the utility model by having sampling task utilities earlier in the tree traversal than seismometer tasks, as shown in Fig. 4a. Using a nonhierarchical utility model, it might be possible to have a large number of seismometer tasks that when summed together produce a higher utility value than a plan with a single sampling task. The hierarchical model does not allow this, as a plan with a sampling task will always dominate a plan with no sampling tasks.

It is important to note that while the concept of dominance establishes a strict preference for some actions over others, it poses no restrictions on the planner to schedule activities in a given order. For example, in a plan with both sampling and seismometer tasks, it does not matter whether the sampling task or seismometer task is performed first. In this way, the planner is allowed to select the best ordering of actions. In practice, the planner tends to schedule the higher priority science objectives first as this is used as a heuristic for the forward search, but this is not a requirement.

This concept of dominance can be extended further to enforce more complex prioritization of tasks. For example, as shown in Fig. 6, it might be desired by the mission designers that collecting one sample is the task of utmost importance, followed by performing a seismometer task. This in turn may be followed by taking a panorama of the surface and then collecting the second and third samples. These

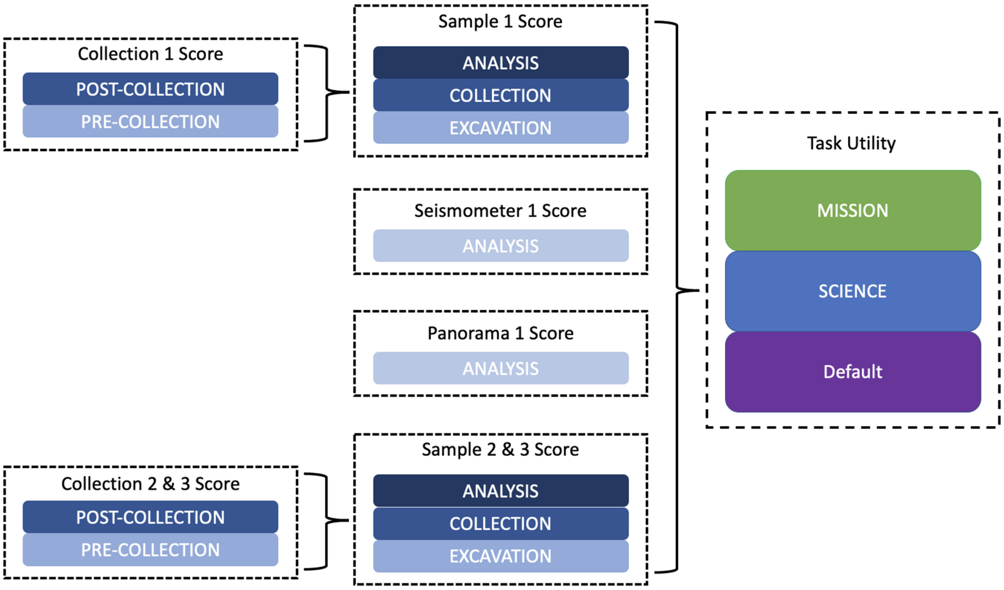



**Fig. 6 Alternative utility model showing how the tree structure can be constructed to account for different science objective orderings. In this example, only the first sample dominates a seismometer task. The seismometer takes precedence over subsequent samples.**

priorities can be translated to the utility model by adding them in the order of dominance, where the first sample is at the highest level, followed by the seismometer at the next, panorama at the third level, and so on. This guarantees that if there is not enough remaining energy to complete all actions, actions will be removed from the plan in order of lowest priority, such as the action to collect samples 2 and 3 will be removed from the plan before removing the first panorama. Because the planners typically schedule highest priority tasks earlier, this also means that samples 2 and 3 will likely be collected after the first sample, seismometer, and panorama data have been collected.

*3. Predicted Utility vs Executed Utility*

After a task has been performed, it is assigned an executed utility. The executed utility is calculated from data products that have actually been generated. It is used to update the utility of a plan in progress. Therefore, the behavior of the Lander may be modified when science activities are executed as the planner can now make more informed decisions based on the collected data.

For planners that perform a forward search, a predicted utility is assigned to tasks that are planned to be executed in the future. We approximate the utility as the average of all data products associated with a particular task utility. Priors are initially used when there is a lack of sensor and execution information at the start of the mission. These priors, initially set by the ground, are updated during the mission using the executed utility. For example, at the beginning of the mission, the planner may be configured with priors to bias the planner to perform sample collection by setting the priors for positive biosignatures to be high. As the Lander performs behaviors, executed utility is updated to reflect the real data product science value rather than the priors. This approach was chosen for its simplicity. While more sophisticated approaches could also be used, such as Bayesian networks or a least-squares approach, defining an accurate system model can be difficult for places like Europa where little is known.

The dominance feature of the utility model biases the planner toward completing the mission objectives even when the predicted science return may be inaccurate. For example, the planner will still attempt to collect samples due to the dominance of sampling over other tasks even if it has little information regarding the expected science return of sampling.

## B. In-Situ Science Collection and Analysis

A primary objective of the Europa Lander Mission Concept is to look for signs of life on Europa. As such, the instruments and corresponding analytics modules must provide an accurate assessment of science instrument data in order to guide the rest of the mission. The data products these modules produce are used to determine which sampling site to go to next and also produce the primary metrics for establishing the success of the mission. The communication constraints of the mission pose several challenges to the instruments. Because of the limited oversight from the ground, the instruments must be robust enough to handle data analysis on their own. Because of the limited communication bandwidth, the instruments must be able to determine what and how much data are needed to send to the ground in order to verify the results of the onboard analysis as not all data can be sent. Both the instruments and analytics modules realize design objective 6, which is to collect data and interpret the findings of any analysis performed on the data.

The instrument simulation modules provide a realistic, stochastic simulation of science instruments that might be required for a given mission. For the purposes of the Europa Lander, the instruments that are currently implemented are categorized as sampling instruments and episodic instruments. The sampling instruments include a gas chromatography–mass spectrometer (GCMS), Raman spectrometer (Raman), and microscope. A sample collected by the Lander is passed to all the sampling instruments, whose respective data, shown in Fig. 7, is sent to the analytics module to look for a positive biosignature. The episodic instruments include a seismometer and a science camera, which are used to detect changes in the environment, such as water plumes.

A major aspect of these simulations is to allow instrument parameters to be dynamically changed in order to simulate different scenarios, such as generating scientifically interesting data or changing the data size and duration. Instrument and science data simulations must produce realistic data product sizes to support realistic downlink planning. All instruments support three predefined scenarios that produce data products of differing science values. To maintain the realistic role separation in the simulation, the science instrument simulation only produces data products and relies on the analysis module to assign an appropriate science value.

The analytics module is an independent module; it is responsible for analyzing data and preprocessing science data for downlinking. For all instruments, the analytics produces one or several science values within the normalized interval [0,1]. The science value is a metric that captures the complexity and number of interesting features of the data [56]. For example, a GCMS-simulated instrument produces science values indicating abundance of chemical compounds and their similarity to a known library of biosignatures. A combination of these science values from all the prototype instruments attributes whether a sample has a high likelihood of a positive biosignature as discussed below.

For the purposes of the Europa Lander, multiple instruments are used to detect positive biosignatures. Multiple instruments are used as a form of redundancy so that false positives and false negatives are limited; only if multiple instruments detect a positive biosignature should the Lander (and ground) consider a sample as actually containing signs of life. To determine how to combine the information taken by multiple instruments to determine a biosignature, we use a chart, shown in Table 1, which describes the set of combined measurements that yield strong evidence of a positive biosignature [5]. The model consists of nine independent lines of evidence shown by the columns in the table. A certain combination of these lines of evidence in a sample produces a positive biosignature. This table is used by the Europa Lander system to score a sample's likelihood of containing a biosignature. Note that the value for each line of evidence is determined by thresholding the science values to obtain a binary 0 or 1 for biosignatures. The threshold can be configured by the ground; however, it is implemented to be 0.5 for the prototype.

In addition to producing a science value for all instruments, the analytics module also provides data reduction and assigns data a downlink priority bin. Data reduction reduces the size of the data by reducing noise while retaining the most interesting features. The degree to which the data are reduced is based on a content-dependent analysis. Data with high science values are generally reduced less than data with low science values as the details are more likely to be interesting to the ground team. The highest level of data reduction returns only summary statistics. A low level of data reduction will return a data product very close to the raw data product. Because this reduction is content dependent and determined by the science objectives of specific instruments, a separate reduction procedure is defined for each instrument. Figure 7 shows examples of raw data products and the associated analysis to obtain the corresponding science value produced by the analytics module. Note that regular data compression, such as GZIP [57], which is not content dependent and can be applied to any data product, is performed later in the pipeline regardless of whether the data product is raw or reduced.

As described in Sec. III.A, after the analytics module generates data products, the science values associated with data products are added to the hierarchical utility model. This step ties together the high-level actions proposed by the planner with the results of executing a particular action. The assessor uses these values determined during execution to predict the quality of science that will be returned from future actions, allowing the planner to consider actions that will maximize the overall expected science return.

## C. Sampling Functional Autonomy

The Lander prototype also includes a sampling autonomy module known as Sampling Autonomy for the Europa Lander (SAEL). While the planning and execution modules perform temporal planning for resource management, communication scheduling, and other higher

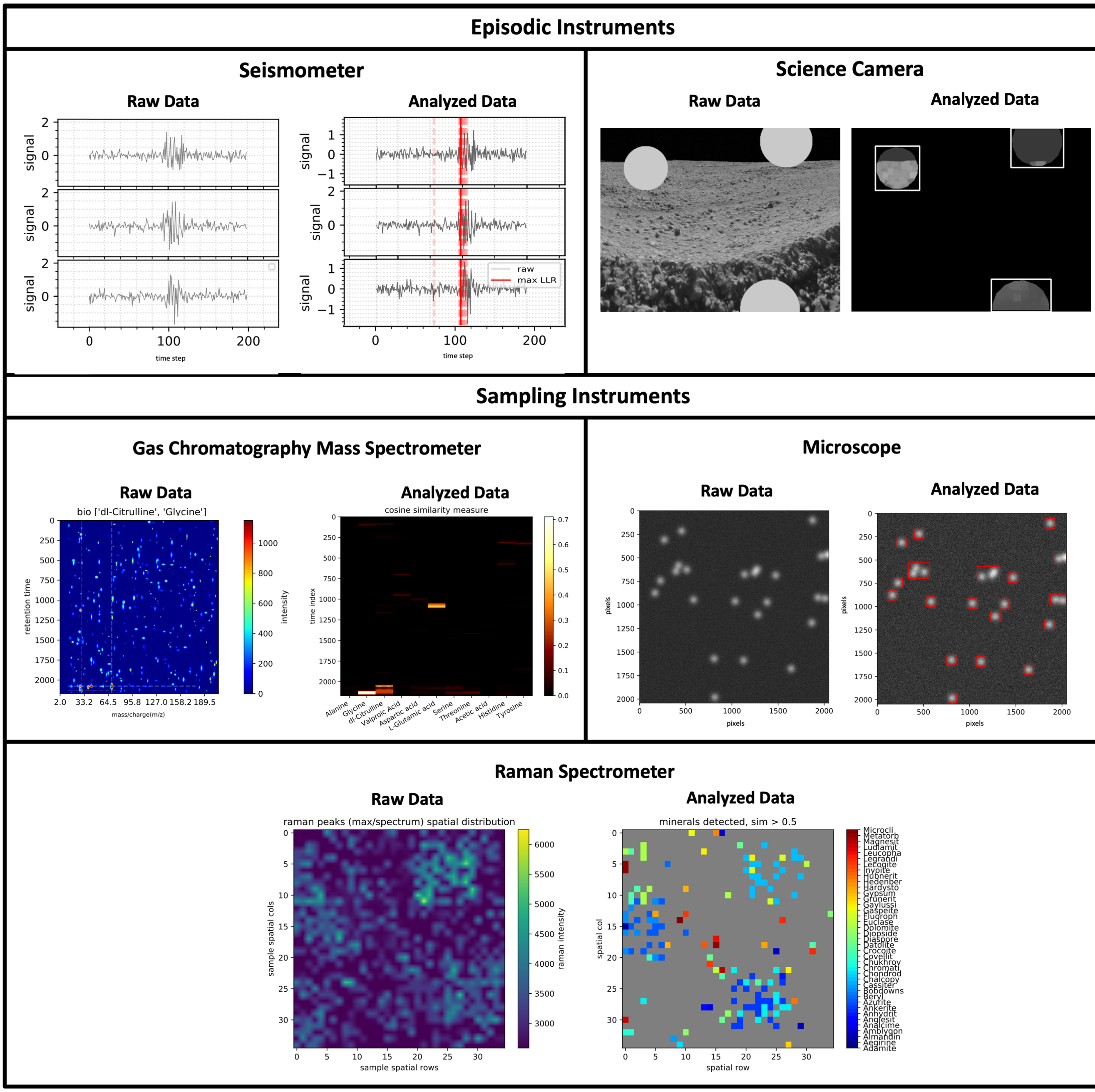



**Fig. 7** Raw data (shown on the left for each instrument example) are analyzed onboard (shown on the right). The analysis is used to calculate science values, and the resulting raw or reduced data product may be downlinked to the ground. The data analysis detects and preserves the interesting characteristics of the raw data product while reducing the overall size of the data product, such as cropping the science camera image shown in the top right to only the interesting sections. Note that the analyzed data shown here are for visualization purposes and demonstrate the detections and other metrics obtained from the analysis, e.g., thumbnailed regions in microscope images (red); the actual data product will likely be a different format.

**Table 1** Biosignature evidence chart from [5] used to fuse information from multiple instruments to determine if a collected sample has a positive or negative biosignature.

| GCMS | | | | Microscope | | Raman | Remote sensing | | |
|---|---|---|---|---|---|---|---|---|---|
| Abundance | Pattern | Chirality | Isotopes | Cell-like structures | Cellular properties | Biominerals | Context | Endo vs Exo | Biosignature |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | Positive |
| 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | Negative |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | Positive |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | Positive |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | Positive |

"Endo" is short for endogenous origin. "Exo" is short for exogenous origin. Each category is populated by the science value from a given sample that is thresholded to be either 0 or 1 in the analytics module.

level activities, SAEL is responsible for identifying suitable excavation sites, adaptively planning tool motions to prepare those sites, and transferring samples to the receiving science instrument. More information about SAEL can be found in [48]. For the purposes of this paper, these activities are treated as black boxes that produce data products that feed into the hierarchical utility model.

### D. Resource Management

Automated resource management is a central concept to the Lander and is implemented through several autonomics modules. Autonomicity can be thought of as "mission self-management" [58], wherein an autonomic module is an isolated, self-governing module that performs a critical set of tasks that are meant to abstract complexity from the rest of the system. The name "autonomic" comes from the "autonomic nervous system" [59] that deals with (continuous) tasks such as breathing or pumping blood. For the Europa Lander, autonomic tasks include energy and thermal management. These tasks are abstracted from the plan, so the planner or executive can focus on "higher-level" tasks. As described by Horn [60], autonomic modules have a detailed knowledge of system components, are aware of and can adapt to the environment, perform some form of self-protection and can recover from faults, and ultimately hide complexity. These features are needed to achieve the design objectives of managing the Lander's finite resources (design objective 4) and reacting to off-nominal conditions (design objective 5). The autonomics produce data products that characterize the state of the Lander and are responsible for some basic safety checks such as turning off a heater that is about to exceed the allowed temperature for a given thermal zone.

#### 1. *Energy Autonomics*

As the Europa Lander Mission Concept is heavily constrained by the remaining energy in the battery, it is important to provide accurate state-of-charge information to the planner. Many external factors can affect the state of charge of the battery, from radiation to temperature changes. Furthermore, the state of charge often cannot be measured directly. The energy autonomics system provides a best estimate based on tracked usage; this value is an uncertain estimate. Dealing with this uncertainty was a major focus of the experimentation. We experimented with starting the mission with a lower state of charge than expected and with the state of charge suddenly dropping to a lower value.

The energy subsystem consists of both an onboard autonomics module that monitors energy usage, and an energy simulation that models hardware components such as current sensors. The onboard module is responsible for keeping track of both the present and historical power. It also tracks energy draw of devices and the activities for which those devices were used. It provides realistic estimates to the planner regarding the state of charge of the battery so that the planner knows how much energy the system has left. It can provide the planner with the expected power draw of individual activities so that the planner can predict how much energy will remain after an activity has completed.

The energy simulation mimics real hardware by publishing the current draw of a device when it is in use. The simulation allows for sensor values and parameters to be dynamically changed in order to simulate different scenarios, such as causing an actuator to draw double the expected amount of current, causing the battery to become significantly depleted due to radiation, or adding sensor noise into the simulation. There is also an interface to support external modules, something used during the field tests described in Sec. V.B to calculate the power draw of the robotic arm.

#### 2. *Thermal Autonomics*

The thermal autonomics consists of both an onboard autonomics module that monitors temperature and controls the heaters, and a thermal simulation that models hardware components such as temperature sensors and heaters.

The onboard module is responsible for maintaining the temperature of predefined thermal zones on the Lander, or regions of the Lander with distinct thermal properties. All devices are expected to have a nominal temperature range at which they should be operated. The ambient temperature of Europa is about 100 K. All hardware, especially actuators, should not be commanded to move due to hardware risk until the zone is at the nominal temperature. The autonomics is responsible for monitoring the temperature of these different zones as well as commanding the heaters to maintain a zone at its nominal temperature. The autonomics is also responsible for maintaining a minimal safe range for all zones when not in use, referred to as survival heating, which ensures that hardware is not damaged when not in use. Survival heating can be disabled, something the ground may choose to do late in the mission in order to conserve more energy at the risk of damaging hardware.

The thermal simulation mimics real hardware by publishing temperature sensor readings and modeling heating using a thermal resistor-capacitor (RC) circuit model for each thermal zone. Several assumptions were made regarding the thermal simulation. As a simplification, we do not model temperature gradients in the actuator lubricant. Furthermore, only a conductive model of heat was used; both convection and radiation were ignored. The time of day was not factored into the ambient temperature, as it was expected that the ambient temperature does not vary extensively on Europa. Parasitic heating was also ignored, such that each zone was considered thermally isolated from other zones. The simulation does consider the impact of devices drawing energy within a thermal zone as a configurable percentage of that energy is lost to heat and thus contributes to the temperature of the zone. This is used to simulate some likely scenarios on the Lander such as the communication modules overheating and requiring a cooldown period if used for too long.

### E. Data Prioritization

For the Europa Lander, the large distance between the Earth and Europa means long light time delays as well as low achievable bandwidth. These difficulties drive the need for an autonomous communications subsystem that is capable of managing when to downlink data, what data should be downlinked, and in what order. The communication stack of the Lander is used to achieve design objective 3, which is to determine what information to send home and when.

#### 1. *Priority*

The system also uses a categorization scheme to classify the relative importance of data products and information. This relative importance is referred to as priority. These priorities specify the order in which data products are downlinked. To determine the relative importance of each data product in the eyes of the ground, four data product priorities are defined along with the rules that detail their required handling:

1) *Transmit Now:* Downlink the data product now, creating a new communication window if necessary.

2) *Decisional:* Downlink the data product at the next available communication window (e.g., scheduled uplink).

3) *Mandatory:* Downlink the data product before end of mission.

4) *Residual:* Downlink the data product only after all other data products have been downlinked.

Note that all downlinks, including for Transmit Now, must schedule downlinks around Europan blackout periods, such as scheduling a downlink as soon as a blackout is over. The priority of the data product will generally be selected according to its science value. For example, on the Europa Lander, if a sample analysis data product has found the presence of a biosignature, it will be marked as Transmit Now as it is important for the ground to receive this information as soon as possible. The mapping between science value and priority is configurable by the ground.

#### 2. *Communication Stack*

The communication stack consists of several components. On the simulation side, a Deep Space Network (DSN) module simulates the radio receiver and transmitter on Earth that talks to the spacecraft. All commands that are sent to the spacecraft go through the DSN module.

The DSN module is connected to a Space Simulation module that simulates the light delay between the spacecraft and Earth. Onboard software includes the communication manager and the downlink manager. The communication manager is used to manage the radio transmitter and receiver to downlink data products and uplink ground information. The downlink manager is used to assign data products to downlink packages.

The downlink manager must balance several requirements when deciding which data products to downlink and when to downlink them. The software supports adding a data priority management table that details how to set the priorities of data products for each data product type. This table selects data product priority given a certain science value. It also specifies what reduction level should be used for each data product. An example configuration is shown in Fig. 8. The priority table is configurable by the ground team.

Priority is not the only constraint in choosing data products to downlink. The downlink must not consume more energy than allocated, it must not overheat the communication hardware, and it must also complete before Earth-set as communication will be lost when the Earth is out of view. The downlink manager selects a subset of existing data products, such that the priority rules and constraints are respected and the number of total downlinked data products is maximized. This is a constrained optimization problem similar to bin-packing, where downlinks are considered to be fixed-sized bins, and data products are fixed-sized items that must fit within those bins. The downlink manager uses Google's OR-Tools [61] as the solver for this optimization problem. The Lander currently supports options to either downlink data at the earliest possible opportunity as well as at the latest possible opportunity. The former is a better risk posture but will consume more energy.

## F. System Execution

The modules involving system execution, namely, the System Behavior Manager (SBM), are used to oversee the execution of the plan generated by the planning layer. The behaviors that the system execution modules perform are used to physically perform all of the design objectives.

### 1. *System Behavior Manager*

To simplify the design of the planners, an abstraction was developed to map high-level tasks to low-level actions. Doing so allows the planners to avoid concerning themselves with low-level details such as heating elements of the hardware before they are used, maintaining that heating during use, and turning heaters off afterward. This abstraction also insulates the planners from changes to the implementation of behaviors.

An example of the mapping between high-level behaviors and low-level actions on the Europa Lander is the "Collect and Transport Sample" behavior. This high-level behavior, performed after an excavation site has been excavated and a sample target has been identified, runs the following low-level actions in order: 1) heat the arm to its nominal temperature; 2) collect the sample target using the arm; 3) capture an image of the collected sample; 4) transport the sample to the onboard instruments; 5) turn of the heat for the arm. The planner or executive requests the high-level behavior, and the system behavior manager commands the low-level actions. These low-level actions can be run sequentially or in parallel. The planners are able to make requests of the low-level parts of the system directly should that be desired; however, they will normally make their requests by requesting the high-level behavior.

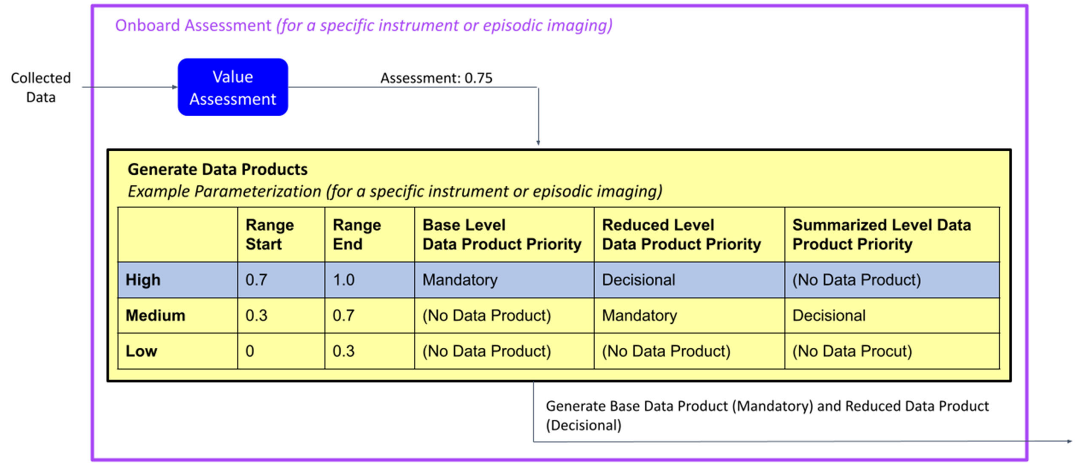
In addition to providing an interface to allow the planners to request high-level behaviors, the SBM also includes the ability to easily redefine the mapping between high-level behaviors and low-level actions during operations. This includes which actions to execute and in what order, what information flows between actions, and how to respond to errors. A companion scripting language describes behaviors in text files, which can be uploaded to the Lander to redefine behaviors. This capability allows the ground to modify the exact implementation of a behavior during the mission to provide fault handling should something go wrong with a behavior.

### 2. *Simulated Clock*

Many of the driving requirements for the Europa Lander center around the limited mission lifespan, and the desire to gather as much science as possible before the battery dies and end of mission is reached. Because of this, many testing and demonstration scenarios involved running the software through the full mission timeframe. To avoid long test times, we decided to simulate these scenarios faster than real time. Originally developed for the CLARAty [35] program, the Simulated Clock provides this capability in ROS using built-in ROS features. The integrated system has many modules that run at different times and have different requirements on how fast (or slow) the clock can run. To coordinate these requirements, the Simulated Clock allows clients to dynamically add and remove a "rate request," which specifies a maximum possible clock speed, and minimum possible clock precision for that client. The Simulated Clock will run at the slowest maximum speed and highest minimum precision for the current set of active rate requests.

### 3. *Hardware Modules*

At the lowest level of the system are hardware modules that carry out behaviors in the real world such as robotic actuators. These modules are controlled by the System Behavior Manager as instructed by the planner. For the Europa Lander, this can include actions such as moving the robotic arm or pointing the high-gain antenna. The software architecture can support various levels of




| | Range Start | Range End | Base Level Data Product Priority | Reduced Level Data Product Priority | Summarized Level Data Product Priority |
|---|---|---|---|---|---|
| **High** | 0.7 | 1.0 | Mandatory | Decisional | (No Data Product) |
| **Medium** | 0.3 | 0.7 | (No Data Product) | Mandatory | Decisional |
| **Low** | 0 | 0.3 | (No Data Product) | (No Data Product) | (No Data Procut) |

**Fig. 8 Onboard assessment diagram shows data product priorities configured by the ground. In this example, if GCMS measurement was assessed a 0.75 science value, two data products would be produced, one with a Base Level reduction and Mandatory priority and another with a Reduced Level reduction and Decisional priority.**

fidelity in these behaviors. We ran experiments with "stubbed" behaviors, which replace the behavior with a wait time, simulators, and full hardware integration. For the purposes of the Europa Lander, there is a separate module known as SAEL that performs autonomy related to this hardware such as trajectory planning for the robotic arm that is described in Sec. III.C.

## IV. Reference Surface Mission

The overall goal of the Europa Lander Mission Concept is to analyze samples collected from the Europan surface, look for signs of biosignatures, and communicate that data back to Earth. There are also several lower priority objectives such as taking panoramic imagery of the Europan surface and collecting seismographic data. Reference Surface Mission (RSM-1) is an example of a mission that accomplishes these objectives. The purpose of the RSM-1 is to explore the flight system and concept of operations (CONOPS) complexities introduced by ground-in-the-loop (GITL) management of initial excavation and sampling activities. During the mission, realistic communication constraints are imposed.

Several mission constraints were chosen to help define the level of autonomy and GITL control that would be tested in RSM-1. For example, sampling activities shall only occur when Earth is in view. This allows for a faster GITL intervention while not mandating GITL control. Additionally, the ground team manages the first three excavation and sampling activities. Autonomy only takes over after these initial activities when ground authorizes the vehicle to do sampling autonomously. Autonomy always manages the episodic activities, such as panoramic imagery and seismometer, and the communication schedule.

The RSM-1 mission flow can be seen in Fig. 9. Our simulation begins after landing on the surface of Europa and performing some initialization activities such as starting the seismometer and episodic panorama data collection. At the beginning of the mission, the Lander does not have the authority to collect samples autonomously. This is done so that the ground can confirm that the Lander is correctly assessing the workspace. At each step within the sampling pipeline, the Lander must stop and wait until it receives the authorization to proceed. The GITL team must assess the Lander's state and use this information to accept or modify future activities. Once the GITL team is satisfied that the Lander autonomy will perform correctly, the team may decide to enable full autonomy that will remove the "stop and wait" points to allow the Lander to sample at will.

As shown in Fig. 10, the sampling pipeline is a loop that repeats for each excavation site. It starts with taking an image of the Lander's workspace to identify potential excavation sites. The imagery is assessed to select the most promising excavation site. The chosen site is excavated. More imagery is taken of the excavated site to identify locations within the site from which to collect samples. The most promising sample is collected and transferred to the science instruments for analysis. RSM-1 defines the order these steps should occur but leaves the planner flexibility in deciding how to schedule them in concert with other activities such as communications and episodic science.

### A. Sample Selection Strategy

For RSM-1, samples are selected according to a sample selection strategy shown in Fig. 11. The strategy imposes several constraints on how samples are collected. The first is that no more than three samples may be collected from a given site. This constraint is imposed to ensure sample variability. The second is that, upon the first negative biosignature at an excavation site, the Lander must collect from a different excavation site. This constraint is imposed to ensure that Lander does not waste its time collecting from a site that is unlikely to have any biosignatures. An example path through site selection, shown in Fig. 11, starts at the site with the highest predicted utility according to the assessor, site A. If the sample does not contain any biosignatures, the Lander switches to site B, which has the next highest utility. Otherwise, the Lander would continue to collect samples at site A. Green arrows show the site transition when a positive biosignature is detected. Red arrows show the site transition when no positive biosignature is detected. This selection criterion is implemented through mission rules as explained in Sec. III.A.

### B. Communicate Until Death

Another requirement of RSM-1 is the communicate-until-death (CUD) transition. This transition occurs at the moment in time when the energy needed to downlink all remaining decisional and mandatory data products onboard the Lander is equal to the amount of energy left in the battery, as shown in Fig. 12. At this point in time, the Lander is expected to stop all activities and spend all the remaining energy transmitting the remaining data products back to Earth. This transition

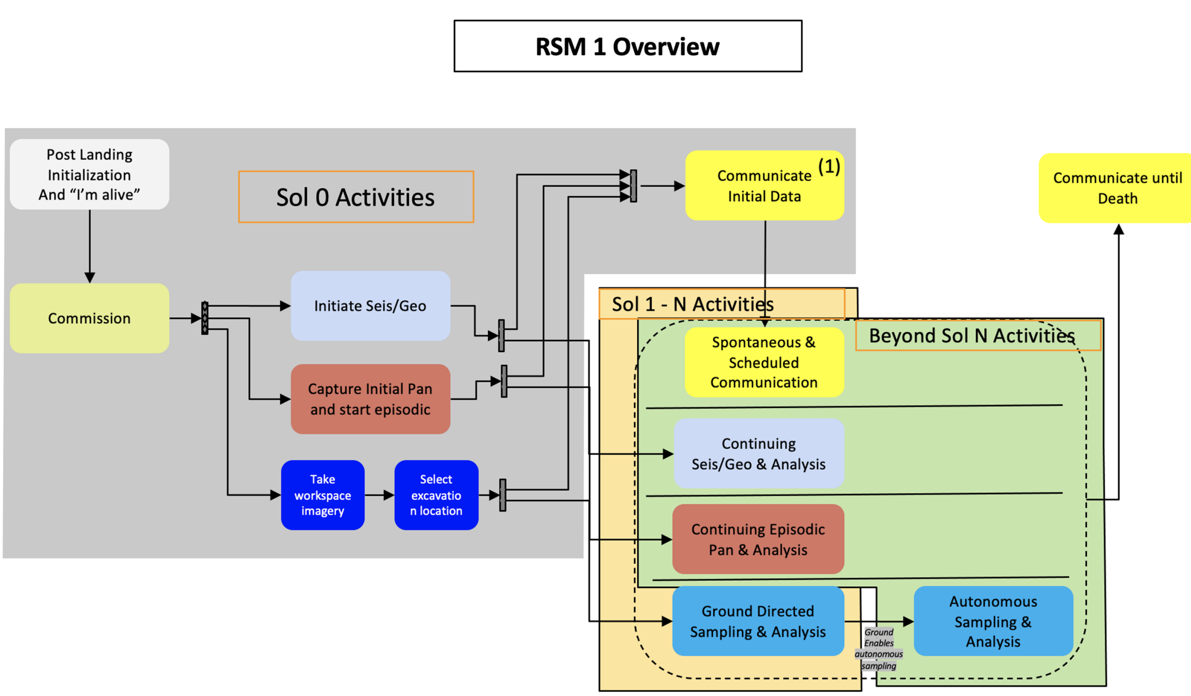



**Fig. 9 This figure shows the Reference Surface Mission 1 overview. During sol 0, commissioning of the spacecraft will be performed, followed by starting the instruments. For the following sols, the Lander will continue to run the instruments while attempting to collect and analyze samples. Note that this is one possible surface mission that could be provided to the planners to plan for and execute.**

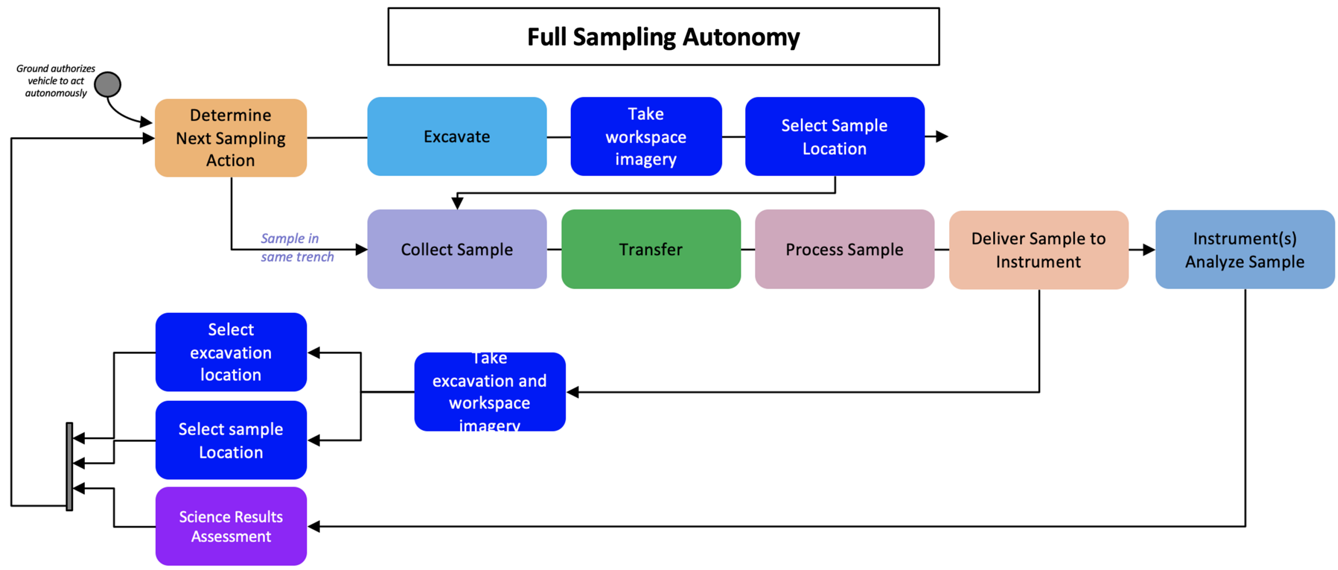



**Fig. 10 This figure shows the steps the Lander will perform when sampling without intervention from the ground. The process includes excavating a site and taking pictures of the site, selecting and collecting a sample from the site, transferring the sample from the arm to the onboard instruments, analyzing the sample, and then repeating the entire process with the prior results factored into future site and sample selections.**

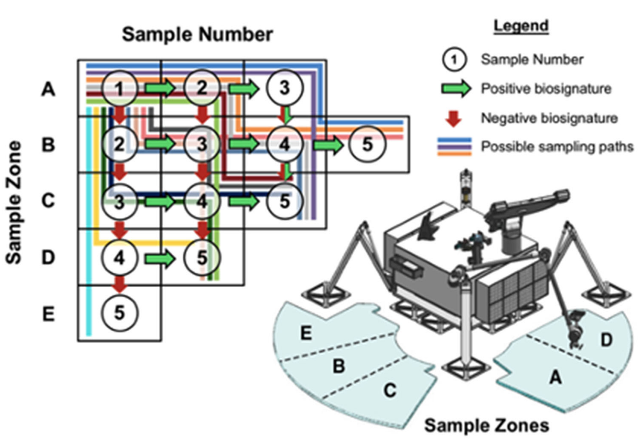



**Fig. 11 Overview of a hypothetical sample selection strategy showing sampling choices for five excavation sites where site A dominates site B, B dominates C, etc. The first negative sample or three positive samples at a given site means that the Lander should switch to a different site.**

is defined so that all required data products make it back to Earth before the Lander dies. Note that residual data products (the lowest priority level) are unlikely to make it back to Earth unless their priority is changed by the ground. The CUD transition highlights the downlink strategy of RSM-1; information that is needed by the GITL to make decisions is transmitted immediately or at the next available opportunity, while data products that are scientifically valuable but not immediately needed are not downlinked until the end of the mission. Transitioning to CUD requires an accurate and robust prediction of when the energy will run out based on the projected activities.

## V. Experimentation

The Europa Lander Autonomy Prototype's main purpose is as an experimentation platform for the Europa Lander Mission Concept. Verification and validation (V&V) is notoriously challenging for most space missions, and will continue to become more challenging as the amount of onboard autonomy increases [62]. These challenges can come from both the technical side, where the autonomy needs to perform the correct actions even in off-nominal situations, as well as on the mission management side, where human mission controllers need to monitor and understand the decisions of the autonomy. It should be noted that the purpose of this paper is to describe the way in which the Europa Lander autonomy architecture enabled sophisticated missions as well as allowed for the evaluation and comparison of two different types of planners for an actual mission. The results shown here are meant to demonstrate the capabilities of the autonomy as demonstrated through scenario-based testing and not provide a recommendation regarding which planner to use for the actual Europa Lander Mission Concept. Instead, this section describes how the autonomy was evaluated for correctness and the metrics that were generated to evaluate the mission-readiness of the planners.

### A. Selected Mission Planners

Two different planning and execution systems were used during experimentation to investigate different planning and execution algorithms. The first chosen system is a utility-based planner called Multimission Executive (MEXEC). MEXEC directly incorporates the hierarchical utility model into its planning environment. The second chosen system is a perspective executive, TRACE, which does not look forward in time but instead reacts to events as they happen. At a high level, TRACE is a much simpler system to use and understand as it is only an executive; its flowchart style plan is defined and uploaded by the ground. MEXEC is a search-based planner that, while generally more difficult to understand than TRACE, is expected to anticipate and react to off-nominal events faster, which will ultimately conserve more energy.

#### *1. MEXEC*

MEXEC is a multimission activity scheduling and execution software that was first created as a prototype demonstration for the Europa Clipper project [63]. MEXEC uses hierarchical task networks to encode abstract representations of command behavior, constraints on timing, and resources usage in order to generate plans. MEXEC performs a forward search where it simulates the impacts of various actions in order to select the best sequence of actions. The best sequence of actions is defined as the one that maximizes utility. For the Lander prototype, this planner was adapted to incorporate the hierarchical utility model described in Sec. III.A. MEXEC uses a branch and bound search to find high utility plans while obeying constraints. MEXEC includes an executive that is responsible for executing each task at its scheduled time, delaying task execution when constraints are not met, monitoring a task while it executes, and aborting a task if constraints fail during execution. MEXEC presents a few challenges for the Europa Lander system such as the difficulties in understanding and visualizing what the planner will do. Furthermore, the search-based approach does not always produce plans the same way a human mission designer might.

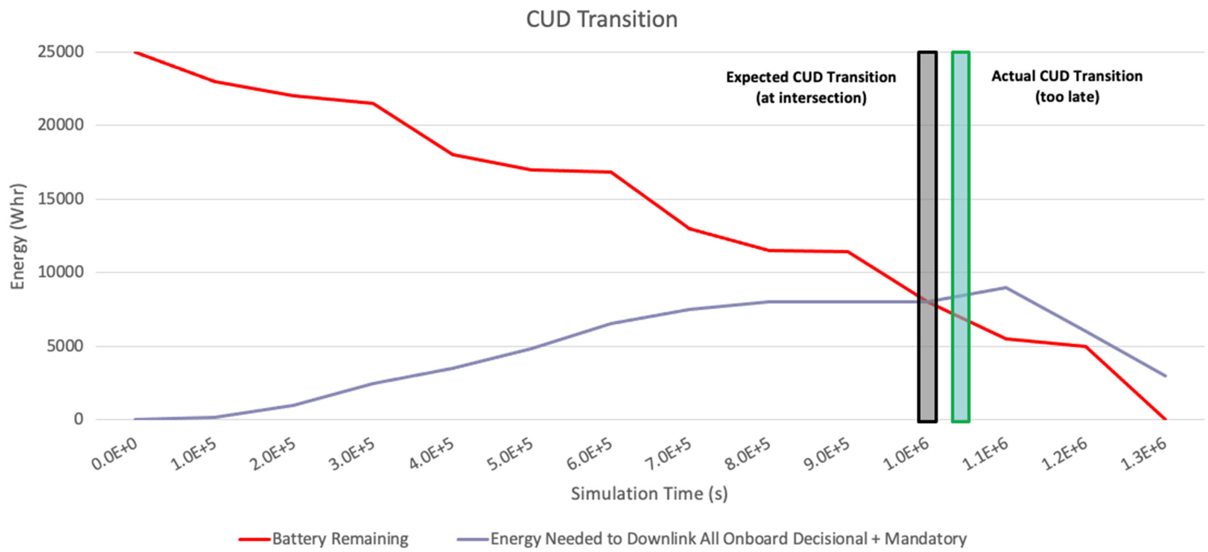



**Fig. 12 Diagram showing the expected communicate until death (CUD) transition compared to the actual CUD transition for a given run. The black bar shows when the system should transition into CUD, while the green bar shows when the actual transition occurred. The transition should occur when the amount of energy left in the battery is equal to the energy needed to downlink all remaining decisional and mandatory data, or the intersection of the red and blue lines. This figure shows an off-nominal case where the transition occurred after the intersection. In a nominal scenario, both the red and blue lines should be equal to zero at the end; the failure shown means that the Lander did not have enough energy to downlink all the remaining data.**



*2. TRACE*

Traceable Robotic Activity Composer and Executive (TRACE) is an onboard executive originally developed for naval applications [64]. TRACE models activities graphically using the Business Process Model and Notation (BPMN) language [65]. These diagrams encapsulate sophisticated, event-driven behavior, especially involving contingency activities to address off-nominal execution. An example diagram is shown in Fig. 13. TRACE is amenable to model checking and formal verification methods. Its use of BPMN allows for very easy understanding of the uploaded plan.

Because TRACE does not plan forward in time, it did not need to make full use of the hierarchical utility model in the same way the MEXEC does. TRACE only cares about the utility for the current state. Because of this, TRACE does not make use of the assessor to produce predicted utility values and instead only considers executed utility values.

Some of the difficulties in using TRACE for the Europa Lander system involve the limited ability to respond to off-nominal events not encapsulated in the BPMN diagram (in other words, a state not defined in the flow chart), the potential limitation of the plans (execution traces) that can be expressed through the BPMN notation, as well as the inability to project resources forward or backward in time, making it difficult to consider the outcomes of particular actions and how that might affect the finite onboard resources.

## B. Mission Management Analysis

To support experimentation, the prototype includes tooling to convey the state and current intent of the onboard autonomy to human operators. Several tools and dashboards were developed to easily run different experiments, visualize the current state of the Lander, uplink new commands and files to the Lander, review the most recent proposed plan generated by onboard autonomy, and postprocess all collected data for a comprehensive mission analysis.

The Lander Dashboard conveys the current state of the Lander, including the temperature of all the thermal zones, which instrument is currently running, and the state of charge of the battery. Most of this information would not be available to the ground team live during operation, but it is a very useful tool for live analysis of the autonomy. The Ground Dashboard displays downlinked data products, such as a panorama of Europa, as well as current science value assessments. It also provides a means to uplink new commands to the Lander.

Finally, there are dashboards to visualize the planners and executives. The TRACE visualization displays a BPMN diagram, similar to that shown in Fig. 13. The visualization showcases the current state of TRACE by highlighting diagram blocks in green to represent where the Lander is in terms of execution. The MEXEC visualization includes a Gantt chart of the current MEXEC plan, including future planned activities. The MEXEC visualization also includes a resource timeline that conveys how MEXEC expects resources to be consumed according to the current plan, such as decreasing the battery state of charge.

## C. Scenario-Based Experimentation

To test the autonomy's behavior in RSM-1, we wished to evaluate how different planners perform when presented with both nominal and off-nominal scenarios. We evaluated the prototype by running 12 different simulated scenarios and comparing each scenario to a

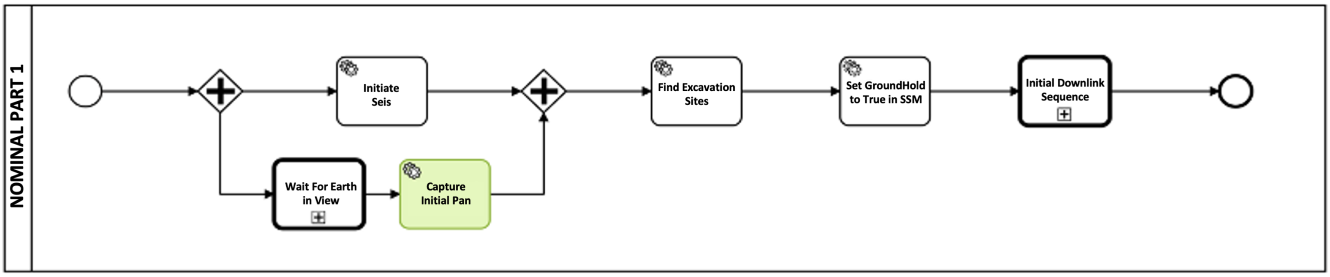



**Fig. 13 BPMN diagram showcasing an example TRACE plan. In this example plan, the Lander will first turn on the seismometer while also taking a panorama when the Earth is in view. It will then identify excavation sites to sample from and later initiate a downlink. BPMN notation allows for abstractions, such as the task "Find Excavation Sites," which are made up of several other primitive tasks.**

**Table 2 Sample of the scenario success criteria**

| Category | Criteria |
|---|---|
| System State | The Lander shall transition to CUD mode. |
| Sampling | The Lander shall choose excavation sites in the order of decreasing pre-excavation value. |
| | The Lander shall follow the science subway map and then collect from all uncollected targets. |
| | The Lander shall perform sample activities only while Earth is in view. |
| Episodic Activities | The Lander shall collect and analyze seismometer data until CUD transition. |
| | The Lander shall take episodic panoramas within specified time intervals until the CUD transition. |
| Communication | The Lander shall downlink and/or uplink only while Earth is in view. |
| | The Lander shall listen for uplinks while downlinking and at each scheduled uplink start time. |
| | The Lander shall immediately initiate a downlink when transmit-now decisional data are generated. |
| | The Lander shall downlink all decisional data products at the next available downlink opportunity. |
| GITL Interaction | The Lander shall wait to receive a ground hold release before resuming the mission. |
| | The Lander shall receive and apply ground overrides to onboard science values and scores. |
| | The Lander shall receive and immediately switch to using any ground-uplinked onboard plans. |

nominal-baseline scenario. To compare scenarios, we established a set of success criteria, examples of which are shown in Table 2. These success criteria were selected as they represent fundamental requirements and constraints of RSM-1 such as listening for uplinks while downlinking, transitioning to CUD at the appropriate time, and choosing excavation sites in the order of highest utility. Successfully achieving each of these fundamental requirements helps prove the capability of the software and shows that it can be used to support additional complexity. Each criterion was evaluated by parsing all the data at the end of a simulated run to ensure that constraints were met and the Lander transitioned to the proper states (such as CUD) at the proper times.

Ideally, for testing, a sensitivity analysis would be performed wherein the various parameters in the system and scenario configuration spaces would be systematically varied. This could then be used to determine the sensitivity (or ideally lack thereof) of the parameters to the environment conditions. Unfortunately, running through the full 30-day mission required over 24 h of simulation time. Therefore, this testing approach was unfeasible. Instead, 12 scenarios were developed that evaluated likely nominal and off-nominal situations for the Lander.

The 12 scenarios are listed in Table 3. These scenarios test different aspects of the system mission objectives, such as changing the

**Table 3 List of the tested scenarios for RSM-1**

| Category | Scenario |
|---|---|
| Science Mission | (Baseline mission) All collected samples have positive biosignatures. |
| | All collected samples have negative biosignatures. |
| | The first collected sample has a negative biosignature. |
| Hardware Fault | Multiple battery packs fail upon landing on Europa. |
| | The battery suddenly depletes midmission. |
| Environmental | The ambient temperature is 15 K lower than expected. |
| Communications | A large amount of decisional data are generated. |
| | A large amount of mandatory data are generated. |
| | The downlink bandwidth suddenly decreases midmission. |
| | The energy allocation for the communications is increased. |
| GITL | Ground overrides the excavation and collection order. |
| | Midmission, the ground uploads an alternate plan/task net for the Lander to execute. |

amount of remaining energy in the battery to evaluate how the Lander manages its finite resources (design objective 4) or producing more data than predicted to see how the Lander will prioritize sending information back home (design objective 3) and maximizing the expected overall science return (design objective 2).

Overall, both planners were able to meet a majority of the scenario success criteria, and they successfully achieved the system design objectives. However, the interesting results are in the details of how they accomplished the RSM-1 mission. By using the prototype as a testbed for different scenarios, we can investigate how changes in the environment or mission rules are interpreted by the autonomy. Due to the large amount of data procured from running these scenarios, we will discuss only some of the results from a single baseline scenario using the TRACE planner and explain the interesting findings from running the off-nominal scenarios with both planners. These selected examples highlight how the prototype can provide insight into the autonomy and help the ground team identify tradeoffs in the mission design.

*1. Baseline Results*

The baseline mission investigates how the RSM-1 mission would be executed under ideal conditions. In this mission, all of the samples collected have positive biosignatures. This means that there is a large amount of scientifically valuable data to downlink. We expect this mission to be executed more quickly than the other scenarios because more valuable data will be generated earlier in the mission. The large amount of data should lead to an earlier CUD transition as well as decreased episodic instrument data collection. For brevity, we will discuss the TRACE results for the baseline mission, but the MEXEC results were functionally the same.

The RSM-1 mission sets a rather conservative communication strategy that attempts to downlink all interesting scientific data before the end of the mission by using a CUD transition. In the baseline mission, the CUD transition occurred after 79.5% of the mission duration had elapsed (state of charge of the battery at the time of transition was 34.2%) as shown in Fig. 14. Based on these results, specifically the need to use more than 20% of the battery to downlink the remaining data products, the ground may consider whether the RSM-1 communication strategy is limiting the overall success of the mission. Instead, they might decide to send fewer data products back to Earth if this means that more science can be achieved.

The hierarchical utility model attempts to maximize the amount of scientific information downlinked by the end of the mission. Throughout the baseline mission, the Lander downlinked 7167 Mbits of data as shown in Fig. 14a, of which about half was decisional and the other half was mandatory. Sampling instrument analysis data accounted for about 70% of the downlinked data, while the sampling perception data accounted for about 26%. Science camera analysis, seismometer instrument analysis, and engineering data products made up the remaining 4%. Because all of the collected samples had positive biosignatures, this mix of science data products is expected. The majority of returned information should involve the successful confirmation of biosignatures, but at the same time the ground should still receive information from the episodic instruments. Other scenarios might prefer a different weighting to these data products depending on the results they contain, such as fewer sampling data products if no biosignatures are found.

The three highest energy consuming onboard activities across the mission were downlink (47.7%), uplink (18.8%), and thermal (13.9%) as shown in Fig. 15. Collectively, these three activities accounted for 80% of the total consumed energy. Downlinking used almost half of the total onboard energy. This is interesting because, in this scenario, the downlink decisions are made by the science autonomy. In a real mission, ground has the option to override the decisions of onboard autonomy and requests data that the Lander has not downlinked. However, this result underscores that downlinking significantly impacts the energy budget of the Lander. If the ground team can trust the autonomy to make optimal downlink decisions, it will likely lead to a more efficient mission.

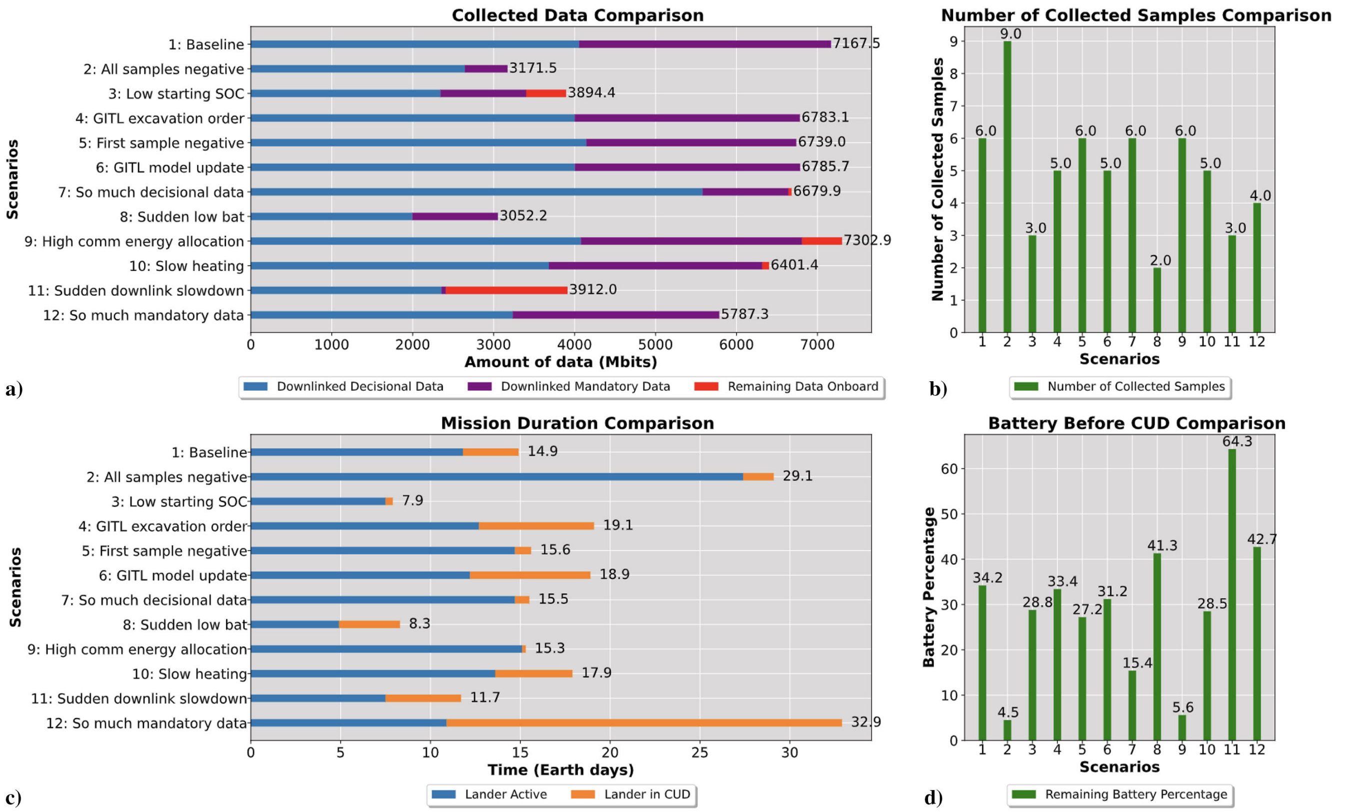



**Fig. 14 Comparison of selected results across all 12 scenarios for TRACE. This figure shows how different factors, such as the environment, hardware failures, and interventions from the ground, impact the mission and overall science return.**

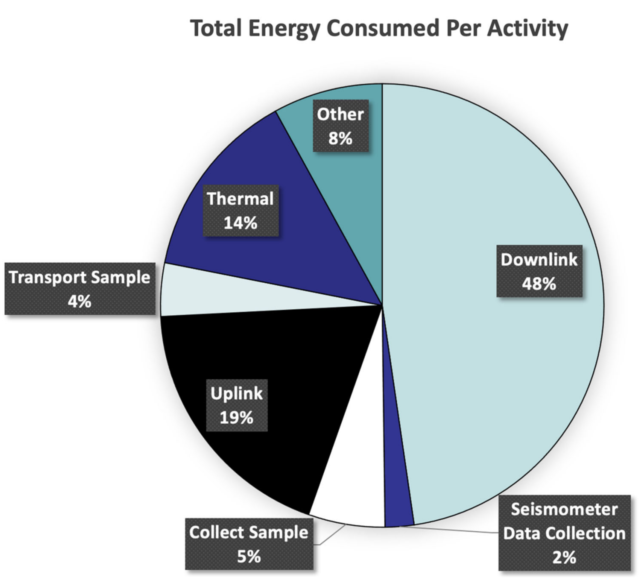



**Fig. 15 Analysis of total onboard energy consumed by each activity during the baseline mission with TRACE, dominated by communication and heating.**

### 2. *Off-Nominal Scenario Results*

Rigorously testing off-nominal scenarios will ensure that the Lander will continue to make acceptable decisions when dealing with unexpected events. Specifically we are interested in events such as the state of charge of the Lander deviating from the expected value, the total onboard data volume deviating from the expected value, unexpected changes to the mission priorities, as well as hardware faults such as several battery packs failing at once. We expect the Lander to recover from these events by adapting to the latest information such as scheduling additional downlinks to handle the increased data volume or decreasing the number of scheduled activities to deal with a decreased state of charge. For faults or unexpected events of a catastrophic nature such as the robotic arm breaking in two, the Lander is expected to wait for instructions from the ground.

Figure 14 shows results from our experimentation using the TRACE planner. One of the largest variations between the baseline and off-nominal scenarios is the number of samples collected as shown in Fig. 14b. The scenarios with a low starting battery, sudden low battery, and sudden downlink slowdown caused the fewest number of samples collected with each collecting two to three samples on average. These results make sense in that due to a lower battery state of charge or slower downlinks requiring additional energy, the number of samples is sacrificed in order to downlink any remaining onboard data. This is yet another depiction of how the ground may want to reprioritize data products in order to perform additional science.

The results of scenario 2, where all collected samples had negative biosignatures, demonstrate the impact of negative biosignatures on the science return of the mission. Figure 14b shows that the Lander collected all nine samples compared to an average of about five samples collected in other scenarios. Because all nine samples were negative, however, the Lander did not have much data to downlink, as shown Fig. 14a. The smaller data volume allows the Lander to live for almost 30 Earth days and transition to CUD much later than the baseline.

Scenario experimentation also confirms that the Lander is able to perform an increased amount of science when operating in the fully autonomous mode compared to when the Lander must wait for instructions from the ground. Scenarios 4 and 6 involved the ground updating the order in which the Lander will excavate sites and replacing the entire onboard plan, respectively. Both scenarios downlinked less data as shown in Fig. 14a and collected fewer samples than the baseline scenario as shown in Fig. 14b. This is likely due to having to wait for the ground to upload instructions. This also shows why the Lander spent more time in CUD than the baseline; waiting for the ground caused the Lander to collect additional science toward the end of the mission. Rather than optimizing when to downlink data, the majority of data had to be downlinked during CUD. The baseline was able to send data down earlier so that there would be less data to downlink at the end and therefore spend less time in CUD. This difference suggests that the autonomy can be trusted to maximize science value and should rarely be overridden by GITL. Both scenarios involved changing the Lander state or plan once. This had a noticeable impact on the mission results, and would be expected to increase if there were additional interventions made by the ground. We believe that continued testing with off-nominal scenarios is a major step in building trust in the autonomy.

We were also very interested to see when CUD occurred in different scenarios, as shown in Fig. 14c, as this affects how many scientific data products are received by ground. Scenario 10, the slow heating scenario, was particularly interesting because it had a later transition to CUD relative to the baseline scenario despite having less energy. It was discovered that sampling tasks, which require the most amount of energy, were not performed as much as in the baseline; sampling tasks produce the largest data products that require the most energy to downlink. By shedding these activities, the planners were able to perform additional episodic data collection (seismometer data, panoramas), which produces relatively small data products that do not require much energy to downlink. As a result, the CUD transition happened later than the baseline.

The results of scenario 11, where the Lander suddenly takes longer to downlink data, demonstrate the impact of exogenous events on the science return of the mission. Figure 14a shows that a lot of data was left onboard the Lander. This makes sense given that the Lander cannot currently anticipate exogenous events and therefore was not able to downlink all the data as planned. Figure 14d shows that the Lander transitioned to CUD with the highest battery percentage out of all the scenarios; this is because the Lander predicted that it would need more energy to handle the suddenly slowed communications. This also caused the Lander to sacrifice collecting additional samples as shown in Fig. 14b.

Comparing the results of scenarios 7 and 12, where the Lander produces additional decisional and mandatory data, respectively, shows the impact of deciding when to schedule downlinks. In the case of decisional data for RSM-1, the Lander is expected to downlink the data at the next available opportunity. By downlinking the decisional data at the next opportunity, the Lander in scenario 7 is still able to collect six samples like in the baseline scenario. Additionally, scenario 7 transitions to CUD later than the baseline, as depicted in Fig. 14c. Note that it does produce significantly less mandatory data than the baseline; this is because it did not collect as much seismometer or panorama data so that it could instead collect more samples. Unlike decisional data, mandatory data are only expected to come down at some point before the end of the mission. Typically, the Lander waits until the CUD transition to downlink all the mandatory data. With this strategy, the Lander collected only four samples and produced significantly less data overall compared to the baseline and scenario 7. It also spent a significant amount of time in CUD as shown in Fig. 14c. This scenario depicts the difficulty in deciding when to transition to CUD as well as the tradeoffs the Lander must consider regarding whether to downlink data it already has or try to collect more science but reserve less energy for downlinking.

Overall it was found that both MEXEC and TRACE showed a sensitivity to the expected duration of activities. In the worst cases, MEXEC and TRACE would delay future activities due to long-running tasks; this could cause the Lander to be late for or altogether miss a scheduled uplink. This also highlighted the need for the ability to pause or cancel certain activities. Pausing activities requires the infrastructure to deconflict resource use, such as the ability to pause the use of the high-gain antenna for taking an episodic panorama and instead use the HGA to downlink data products.

Individually, TRACE was relatively easy to use as well as understand the progress of TRACE in the mission. It was found that TRACE will ultimately need a forward prediction capability in order to better model the resources of the system as it is currently purely reactive. For example, TRACE would sometimes run an activity that produced a large data product right before the CUD transition. It did not predict that producing a large data product right before the CUD transition would cause the Lander to now have insufficient energy to downlink all onboard data products.

It was found that MEXEC had a strong ability to respond to unplanned events during the mission. For example, in one scenario, MEXEC started the transition to CUD earlier than it needed to (this was due to an overly conservative energy estimate). After all data products had been downlinked, MEXEC saw that it had remaining energy onboard and decided to transition out of CUD and collect more data. This behavior was not explicitly defined and shows the potential benefits of a utility-based planner. Despite this potential benefit, a major challenge with MEXEC was understanding why the planner chose to perform certain actions. Specifically, it was not always clear how changing certain utility values would affect the overall mission.

### D. Alaska Field Test

Experimentation using our autonomy on actual hardware has already begun. In June 2022, an operational readiness test (ORT) of a prototype Lander with an attached arm was performed. Photographs of this test are shown in Fig. 16. Although the primary focus of the test was validation of the SAEL framework to properly perform excavations and sample collections, the autonomy software using the TRACE executive commanded the mission. Following the success of the ORT, the prototype Lander and arm participated in a field test demonstration in August 2022 on the Matanuska Glacier in Alaska as shown in Fig. 1. Bowkett et al. [66] present the results from this test. The field test, using SAEL, the autonomy software, and TRACE, was

**Fig. 16 Prototype Lander used during operational readiness test.**

designed to showcase the progression of activities that constitute sample acquisition. Activities included the use of perception to discriminate site selection, performing a site excavation with an attached drill, and performing a sample collection from an excavated site using an attached scoop. During the field test, the Lander successfully demonstrated end-to-end sampling where it excavated 10 cm into the surface and collected 100 g of subsurface material. The Alaska field test also used the TRACE executive as part of the 58 different sampling tests performed on the glacier. For more details, please refer to the referenced paper.

## VI. Future Work

The purpose of the Europa Lander Autonomy Prototype is to strengthen the understanding of autonomy design in different mission scenarios. As such, it has helped us identify a number of improvements that can be made to the planners and the utility model.

One of the key areas of the hierarchical utility model that should be investigated next is incorporating uncertainty into the model. Doing so will help make the system more robust to uncertainty such as stochastic activity durations; if a task seems promising but is unlikely to succeed given its constraints, the planner may decide on a different course of action. Machine learning algorithms should be investigated to further improve the capabilities of the autonomy, from modeling activity durations to forecasting events, such as an erupting geyser, so that the Lander is prepared to take a picture. The limited knowledge of Europa, in addition to the need to explain results to mission operations, will make this a very difficult but interesting challenge.

Our results suggest that the explainability of the planners can be further improved. Although the hierarchical utility model can convey intent within the system, it does not answer questions such as why the planner did not choose alternative actions. For a mission like the Europa Lander that heavily uses autonomy, mission operators will need to be able to trust the autonomy to make correct decisions. As we have shown in this paper, running various experiments with off-nominal situations can build confidence in the autonomy, but it does not yet reach the level of trust desired by mission controllers. To reach this level will require greater explainability within the system. This includes a means to communicate not just the choices made but the alternatives that were considered. This also includes compact ways to better represent both the current and future set of actions chosen by the planner.

This work can also be extended to benefit other missions. A focus of this work has been to create a reusable architecture that will work for more missions than just the Europa Lander. To that end, the software is highly modular such that different components can be replaced with ease.

## VII. Conclusions

In this paper, we present an autonomous software prototype that can execute complex and highly constrained missions with limited interventions from humans. We define six design objectives that are critical to the success of a Europa Lander Mission Concept, and evaluate our design based upon its ability to achieve these objectives. Many of these design objectives are achieved by using a hierarchical utility model. This model allows for various information and data products produced by the system to be grouped together and reasoned about at levels of abstraction. The various levels of abstraction are used by the Lander to fulfill all of its system design objectives, namely, maximizing the number of completed mission objectives, maximizing the expected overall science return, managing finite resources, reacting to off-nominal events, collecting data and interpreting findings in situ, and finally determining what information to send back to Earth and when. These objectives represent several of the major challenges the Lander will face on Europa, such as a limited lifetime, extreme uncertainty, and communication blackouts.

We further present a means to prioritize which data products should be downlinked first and describe an architecture for in-situ analysis of science data products. We show how the priority of data is based on the information contained within the data product. We evaluate the scientific value through both in-situ analysis using the onboard instruments as well as a predicting scientific value based on past measurements. These features allow the Lander to make decisions about the quality of science performed, which in turn is used to inform future actions.

We demonstrate our success in achieving the system design objectives through several different scenario-based tests run both in simulation and on actual hardware as well as from different users studies regarding the behaviors for the autonomy. These studies highlight the capabilities and importance of autonomy for decision making, which if performed on Earth instead of Europa, would lead to a vast reduction in the total amount of science performed.

## Acknowledgments

The research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with NASA. We would like to thank everyone on the 30+ Europa Lander Autonomy Prototype team from the Jet Propulsion Laboratory. We would not have been able to perform this development or experiments without their expertise and help. In particular, we would like to acknowledge, in no particular order, Jeffrey Edlund, William Seto, Yumi Iwashita, Todd Litwin, Igor Uchenik, Hrand Aghazarian, Garrett Johnson, Dan Delany, Jeff Pamer, Jacob Kosberg, Peter di Pasquale, Anna Sabel, Gregg Rabideau, JP De la Croix, Grace Lim, Jenette Sellin, David Kim, Brett Kennedy, Grace Tan-Wang, Vandi Verma, Paul Backes, and Joseph Bowkett.

## References

[1] National Research Council, *Vision and Voyages for Planetary Science in the Decade 2013–2022*, National Academies Press, Washington, D.C., 2011, Chap. 11, https://nap.nationalacademies.org/catalog/13117/vision-and-voyages-for-planetary-science-in-the-decade-2013-2022. https://doi.org/10.17226/13117

[2] National Academies of Sciences, Engineering, and Medicine, *Origins, Worlds, and Life: A Decadal Strategy for Planetary Science and Astrobiology 2023–2032*, National Academies Press, Washington, D.C., 2022, Chap. 21, https://nap.nationalacademies.org/catalog/26522/origins-worlds-and-life-a-decadal-strategy-for-planetary-science. https://doi.org/10.17226/26522

[3] Hays, L., Archenbach, L., Bailey, J., Barnes, R., Barros, J., Bertka, C., Boston, P., Boyd, E., Cable, M., Chen, I., et al., *NASA Astrobiology Strategy*, NASA, Washington, D.C., 2015, Chap. 5.4.2.

[4] National Academies of Sciences, Engineering, and Medicine, *An Astrobiology Strategy for the Search for Life in the Universe*, National Academies Press, Washington, D.C., 2019, Chap. 5.

[5] Hand, K. P., "Report of the Europa Lander Science Definition Team," NASA JPL D-97667, 2017, https://europa.nasa.gov/resources/58/europa-lander-study-2016-report/.

[6] Dooley, J., "Mission Concept for a Europa Lander," *2018 IEEE Aerospace Conference*, Inst. of Electrical and Electronics Engineers, New York, 2018, pp. 1–10. https://doi.org/10.1109/AERO.2018.8396518

[7] "Europa," NASA, June 2023, https://solarsystem.nasa.gov/moons/jupiter-moons/europa/in-depth/.

[8] "Europa Clipper: Exploring Jupiter's Icy Moon," NASA, June 2014, https://europa.nasa.gov/.

[9] Gaines, D. M., Anderson, R. C., Doran, G. B., Huffman, W., Justice, H., Mackey, R. M., Rabideau, G. R., Vasavada, A. R., Verma, V., Estlin, T. A., Fesq, L. M., Ingham, M. D., Maimone, M. W., and Nesnas, I. A. D., "Productivity Challenges for Mars Rover Operations," Jet Propulsion Lab., NASA JPL-CL-16-2123, 2016, https://dataverse.jpl.nasa.gov/dataset.xhtml?persistentId=hdl:2014/46121.

[10] Gaines, D., Doran, G., Paton, M., Rothrock, B., Russino, J., Mackey, R., Anderson, R., Francis, R., Joswig, C., Justice, H., Kolcio, K., Rabideau, G., Schaffer, S., Sawoniewicz, J., Vasavada, A., Wong, V., Yu, K., and Agha-mohammadi, A.-A., "Self-Reliant Rovers for Increased Mission Productivity," *Journal of Field Robotics*, Vol. 37, No. 7, 2020, pp. 1171–1196, https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21979. https://doi.org/10.1002/rob.21979

[11] Chattopadhyay, D., Mishkin, A., Allbaugh, A., Cox, Z. N., Lee, S. W., Tan-Wang, G., and Pyrzak, G., "The Mars Science Laboratory Supratactical Process," *SpaceOps 2014 Conference*, AIAA Paper 2014-1940, 2014. https://doi.org/10.2514/6.2014-1940

[12] Theiling, B., Brinckerhoff, W., Castillo-Rogez, J., Chou, L., Poian, V. D., Graham, H., Hosseini, S. S., Lyness, E., MacKinnon, J., Neveu, M., Raimalwala, K., and Thompson, B., "Non-Robotic Science Autonomy Development," *Bulletin of the AAS*, Vol. 53, No. 4, 2021, p. 48, https://baas.aas.org/pub/2021n4i048.

[13] Verma, V., and Kuhn, S., "Refactoring the Curiosity Rover's Sample Handling Architecture on Mars," *2019 IEEE Aerospace Conference*, Inst. of Electrical and Electronics Engineers, New York, 2019, pp. 1–12. https://doi.org/10.1109/AERO.2019.8741988

[14] Verma, V., Carsten, J., and Kuhn, S., "The Evolution of the Curiosity Rover Sampling Chain," *Journal of Field Robotics*, Vol. 37, No. 5, 2020, pp. 729–753. https://doi.org/10.1002/rob.21913

[15] Arvidson, R. E., Bonitz, R. G., Robinson, M. L., Carsten, J. L., Volpe, R. A., Trebi-Ollennu, A., Mellon, M. T., Chu, P. C., Davis, K. R., Wilson, J. J., Shaw, A. S., Greenberger, R. N., Siebach, K. L., Stein, T. C., Cull, S. C., Goetz, W., Morris, R. V., Ming, D. W., Keller, H. U., Lemmon, M. T., Sizemore, H. G., and Mehta, M., "Results from the Mars Phoenix Lander Robotic Arm experiment," *Journal of Geophysical Research: Planets*, Vol. 114, No. E1, 2009. https://doi.org/10.1029/2009 JE003408

[16] Bonitz, R., Shiraishi, L., Robinson, M., Carsten, J., Volpe, R., Trebi-Ollennu, A., Arvidson, R. E., Chu, P. C., Wilson, J. J., and Davis, K. R., "The Phoenix Mars Lander Robotic Arm," *2009 IEEE Aerospace Conference*, Inst. of Electrical and Electronics Engineers, New York, 2009, pp. 1–12. https://doi.org/10.1109/AERO.2009.4839306

[17] Boynton, W. V., Feldman, W. C., Squyres, S. W., Prettyman, T. H., Brückner, J., Evans, L. G., Reedy, R. C., Starr, R., Arnold, J. R., Drake, D. M., Englert, P. A. J., Metzger, A. E., Mitrofanov, I., Trombka, J. I., d'Uston, C., Wänke, H., Gasnault, O., Hamara, D. K., Janes, D. M., Marcialis, R. L., Maurice, S., Mikheeva, I., Taylor, G. J., Tokar, R., and Shinohara, C., "Distribution of Hydrogen in the Near Surface of Mars: Evidence for Subsurface Ice Deposits," *Science*, Vol. 297, No. 5578, 2002, pp. 81–85. https://doi.org/10.1126/science.1073722

[18] Moore, H. J., and Jakosky, B. M., "Viking Landing Sites, Remote-Sensing Observations, and Physical Properties of Martian Surface Materials," *Icarus*, Vol. 81, No. 1, 1989, pp. 164–184, https://www.sciencedirect.com/science/article/pii/0019103589901322. https://doi.org/10.1016/0019-1035(89)90132-2

[19] Abarca, H., Deen, R., Hollins, G., Zamani, P., Maki, J., Tinio, A., Pariser, O., Ayoub, F., Toole, N., Algermissen, S., Soliman, T., Lu, Y., Golombek, M., Calef, F., Grimes, K., De Cesare, C., and Sorice, C., "Image and Data Processing for InSight Lander Operations and Science," *Space Science Reviews*, Vol. 215, March 2019, pp. 1–53. https://doi.org/10.1007/s11214-019-0587-9

[20] Trebi-ollennu, A., Kim, W., Ali, K., Khan, O., Sorice, C., Bailey, P., Umland, J., Bonitz, R., Ciarleglio, C., Knight, J., Haddad, N., Klein, K., Nowak, S., Klein, D., Onufer, N., Glazebrook, K., Kobeissi, B., Baez, E., Sarkissian, F., and Lin, J., "InSight Mars Lander Robotics Instrument Deployment System," *Space Science Reviews*, Vol. 214, Aug. 2018, pp. 1–18. https://doi.org/10.1007/s11214-018-0520-7

[21] Rabideau, G., and Benowitz, E., "Prototyping an Onboard Scheduler for the Mars 2020 Rover," *International Workshop on Planning and Scheduling for Space (IWPSS 2017)*, Steve Chien of Jet Propulsion Laboratory, Caltech Inst. of Technology and Sean Augenstein of Google, Pittsburgh, CA, 2017, https://ai.jpl.nasa.gov/public/papers/rabideau_iwpss2017_prototyping.pdf.

[22] Grip, H. F., Lam, J., Bayard, D. S., Conway, D. T., Singh, G., Brockers, R., Delaune, J. H., Matthies, L. H., Malpica, C., Brown, T. L., and Jain, A., "Flight Control System for NASA's Mars Helicopter," *AIAA Scitech 2019 Forum*, AIAA Paper 2019-1289, 2019.

[23] Balaram, J., Aung, M., and Golombek, M. P., "The Ingenuity Helicopter on the Perseverance Rover," *Space Science Reviews*, Vol. 217, No. 4, 2021, pp. 1–11. https://doi.org/10.1007/s11214-021-00815-w

[24] Ono, M., Doran, G., Langert, E., Wagstaff, K., Kim, D. I., Gaut, A., Estlin, T., Jain, A., Cameron, M., Muliere, D., Roberts, E., Kriechbaum, K., and Reeves, G., "ARIEL: Autonomous Excavation Site Selection for Europa Lander Mission Concept," *2020 IEEE Aerospace Conference*, Inst. of Electrical and Electronics Engineers, New York, 2020, pp. 1–19. https://doi.org/10.1109/AERO47225.2020.9172744

[25] Shaw, A., Arvidson, R. E., Bonitz, R., Carsten, J., Keller, H. U., Lemmon, M. T., Mellon, M. T., Robinson, M., and Trebi-Ollennu, A., "Phoenix Soil Physical Properties Investigation," *Journal of Geophysical Research: Planets*, Vol. 114, No. E1, 2009, pp. 29–48. https://doi.org/10.1029/2009 JE003455

[26] Estlin, T. A., Bornstein, B. J., Gaines, D. M., Anderson, R. C., Thompson, D. R., Burl, M., Castaño, R., and Judd, M., "AEGIS Automated Science Targeting for the MER Opportunity Rover," *ACM Transactions on Intelligent Systems and Technology (TIST)*, Vol. 3, No. 3, 2012, pp. 1–19. https://doi.org/10.1145/2168752.2168764

[27] Francis, R., Estlin, T., Doran, G., Johnstone, S., Gaines, D., Verma, V., Burl, M., Frydenvang, J., Montaño, S., Wiens, R. C., Schaffer, S., Gasnault, O., DeFlores, L., Blaney, D., and Bornstein, B., "AEGIS Autonomous Targeting for ChemCam on Mars Science Laboratory: Deployment and Results of Initial Science Team Use," *Science Robotics*, Vol. 2, No. 7, 2017, Paper eaan4582. https://doi.org/10.1126/scirobotics.aan4582

[28] Chien, S., Mclaren, D., Tran, D., Davies, A. G., Doubleday, J., and Mandl, D., "Onboard Product Generation on Earth Observing One: A Pathfinder for the Proposed Hyspiri Mission Intelligent Payload Module," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, Vol. 6, No. 2, 2013, pp. 257–264. https://doi.org/10.1109/JSTARS.2013.2249574

[29] Chien, S., Doubleday, J., Thompson, D. R., Wagstaff, K. L., Bellardo, J., Francis, C., Baumgarten, E., Williams, A., Yee, E., Stanton, E., and Piug-Suari, J., "Onboard Autonomy on the Intelligent Payload Experiment Cubesat Mission," *Journal of Aerospace Information Systems*, Vol. 14, No. 6, 2017, pp. 307–315. https://doi.org/10.2514/1.I010386

[30] Verma, V., Gaines, D., Rabideau, G., Schaffer, S., and Joshi, R., "Autonomous Science Restart for the Planned Europa Mission with Lightweight Planning and Execution," Jet Propulsion Lab., NASA JPL-CL-18-0417, 2017, https://dataverse.jpl.nasa.gov/dataset.xhtml?persistentId=hdl:2014/48030.

[31] Yelamanchili, A., Agrawal, J., Chien, S., Biehl, J., Connell, A., Guduri, U., Hazelrig, J., Ip, I., Maxwell, K., Steadman, K., and Towey, S., "Ground-Based Automated Scheduling for the Mars 2020 Rover," *Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation for Space*, ESA, Noordwijk, The Netherlands, 2020, https://ai.jpl.nasa.gov/public/papers/M2020-Ground-i-SAIRAS2020-camera.pdf.

[32] Chien, S., Bue, B., Castillo-Rogez, J., Gharibian, D., Knight, R., Schaffer, S., Thompson, D., and Wagstaff, K., "Agile Science: Onboard Autonomy for Primitive Bodies and Deep Space Exploration," *13th International Conference on Space Operations, SpaceOps 2014*, AIAA Paper 2014-1888, 2014.

[33] Rabideau, G., Wong, V., Gaines, D., Agrawal, J., Chien, S., Kuhn, S., Fosse, E., and Biehl, J., "Onboard Automated Scheduling for the Mars 2020 Rover," *Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation for Space*, ESA, Noordwijk, The Netherlands, 2020, https://ai.jpl.nasa.gov/public/papers/M2020-OBP-i-SAIRAS2020-camera.pdf.

[34] Amini, R., Fesq, L., Mackey, R., Mirza, F., Rasmussen, R., Troesch, M., and Kolcio, K., "FRESCO: A Framework for Spacecraft Systems Autonomy," *2021 IEEE Aerospace Conference (50100)*, Inst. of Electrical and Electronics Engineers, New York, 2021, pp. 1–18. https://doi.org/10.1109/AERO50100.2021.9438470

[35] Volpe, R., Nesnas, I., Estlin, T., Mutz, D., Petras, R., and Das, H., "The CLARAty Architecture for Robotic Autonomy," *2001 IEEE Aerospace Conference Proceedings (Cat. No.01TH8542)*, Vol. 1, Inst. of Electrical and Electronics Engineers, New York, 2001, pp. 1/121–1/132. https://doi.org/10.1109/AERO.2001.931701

[36] Cashmore, M., Fox, M., Long, D., Magazzeni, D., Ridder, B., Carrera, A., Palomeras, N., Hurtos, N., and Carreras, M., "ROSPlan: Planning in the Robot Operating System," *Proceedings of the International Conference on Automated Planning and Scheduling*, Vol. 25, No. 1, Association for the Advancement of Artificial Intelligence (AAAI) Press, Washington, D.C., 2015, pp. 333–341, https://ojs.aaai.org/index.php/ICAPS/article/view/13699. https://doi.org/10.1609/icaps.v25i1.13699

[37] Barreiro, J., Boyce, M. E., Do, M. B., Frank, J. D., Iatauro, M., Kichkaylo, T., Morris, P. H., Ong, J. C., Remolina, E., Smith, T. B., and Smith, D. E., "EUROPA: A Platform for AI Planning, Scheduling, Constraint Programming, and Optimization," *4th International Competition on Knowledge Engineering for Planning and Scheduling (ICKEPS)*, ICAPS, 2012, pp. 6–7.

[38] Niemueller, T., Hofmann, T., and Lakemeyer, G., "Goal Reasoning in the CLIPS Executive for Integrated Planning and Execution," *Proceedings of the International Conference on Automated Planning and Scheduling*, Vol. 29, No. 1, Association for the Advancement of Artificial Intelligence (AAAI) Press, Washington, D.C., 2021, pp. 754–763, https://ojs.aaai.org/index.php/ICAPS/article/view/3544. https://doi.org/10.1609/icaps.v29i1.3544

[39] Verma, V., Jonsson, A., Pasareanu, C., and Iatauro, M., "Universal-Executive and PLEXIL: Engine and Language for Robust Spacecraft Control and Operations," *Space 2006*, AIAA Paper 2006-7449, 2006, https://doi.org/10.2514/6.2006-7449

[40] Williams, B., Ingham, M., Chung, S., and Elliott, P., "Model-Based Programming of Intelligent Embedded Systems and Robotic Space Explorers," *Proceedings of the IEEE*, Vol. 91, No. 1, 2003, pp. 212–237.
https://doi.org/10.1109/JPROC.2002.805828

[41] Muscettola, N., Nayak, P., Pell, B., and Williams, B. C., "Remote Agent: To Boldly Go Where No AI System has Gone Before," *Artificial Intelligence*, Vol. 103, No. 1, 1998, pp. 5–47, https://www.sciencedirect.com/science/article/pii/S000437029800068X.
https://doi.org/10.1016/S0004-3702(98)00068-X

[42] Musliner, D. J., and Goldman, R. P., "Priority-Based Meta-Control within Hierarchical Task Network Planning," *2010 Fourth IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshop*, Inst. of Electrical and Electronics Engineers, New York, 2010, pp. 282–286.
https://doi.org/10.1109/SASOW.2010.58

[43] Roijers, D. M., and Whiteson, S., *Multi-Objective Decision Problems*, Springer International Publishing, Cham, Switzerland, 2017, pp. 9–17.
https://doi.org/10.1007/978-3-031-01576-2_2

[44] Wellman, M. P., and Doyle, J., "Modular Utility Representation for Decision-Theoretic Planning," *Artificial Intelligence Planning Systems*, edited by J. Hendler, Morgan Kaufmann, San Francisco, 1992, pp. 236–242, https://www.sciencedirect.com/science/article/pii/B9780080499444500331.
https://doi.org/10.1016/B978-0-08-049944-4.50033-1

[45] Basich, C., Russino, J. A., Chien, S., and Zilberstein, S., "A Sampling Based Approach to Robust Planning for a Planetary Lander," *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Inst. of Electrical and Electronics Engineers, New York, 2022, pp. 4106–4111.
https://doi.org/10.1109/IROS47612.2022.9981083

[46] Lekavý, M., and Návrat, P., "Expressivity of STRIPS-Like and HTN-Like Planning," *Agent and Multi-Agent Systems: Technologies and Applications*, edited by N. T. Nguyen, A. Grzech, R. J. Howlett, and L. C. Jain, Springer, Berlin, 2007, pp. 121–130.
https://doi.org/10.1007/978-3-540-72830-6_13

[47] Erol, K., Hendler, J., and Nau, D., "HTN Planning: Complexity and Expressivity," *Proceedings of the National Conference on Artificial Intelligence*, Vol. 2, Assoc. for the Advancement of Artificial Intelligence (AAAI) Press, Washington, D.C., 1994, pp. 1123–1128.

[48] Bowkett, J., Nash, J., Kim, D. I., Kim, S.-K., Thakker, R., Brinkman, A., Cheng, Y., Willson, R., Lim, C., Gaut, A., Jain, A., Gildner, M., Emanuel, B., and Backes, P., "Functional Autonomy Challenges in Sampling for an Europa Lander Mission," *2021 IEEE Aerospace Conference (50100)*, Inst. of Electrical and Electronics Engineers, New York, 2021, pp. 1–8.
https://doi.org/10.1109/AERO50100.2021.9438298

[49] Kim, S. Y., Roffo, K., Ye, S. C., Tan-Wang, G., Laubach, S., Pyrzak, G., and Reeves, G., "Designing for Operating Autonomous Space Missions: Concept of Operations Design for Europa Lander Using Mission-Level Modeling and Simulation," *ASCEND 2021*, AIAA Paper 2021-4116, 2021.

[50] Lattimore, M., Karban, R., Gomez, M. P., Bovre, E., and Reeves, G. E., "A Model-Based Approach for Europa Lander Mission Concept Exploration," *2022 IEEE Aerospace Conference (AERO)*, Inst. of Electrical and Electronics Engineers, New York, 2022, pp. 1–13.
https://doi.org/10.1109/AERO53065.2022.9843241

[51] Nayar, H., Goel, A., Boettcher, A., Hans, M., Sawoniewicz, J., Gaut, A., Higa, S., Ono, H., Jain, A., Lim, C., Nesnas, I., Ma, R., and Thomsen, M., "Development of a Lander Autonomy Testbed for Ocean Worlds Missions," *Earth and Space 2021*, American Soc. of Civil Engineers (ASCE) Library, Reston, VA, 2021, pp. 531–540.
https://doi.org/10.1061/9780784483374.050

[52] Edwards, L. J., Wong, U. Y., Dalal, K. M., Kulkarni, C. S., Rogg, A., Tardy, A., Stucky, T. R., Umurhan, O. M., Catanoso, D., and Welsh, T. M., "An Autonomy Software Testbed Simulation for Ocean Worlds Missions," *Earth and Space 2021*, American Soc. of Civil Engineers (ASCE) Library, Reston, VA, 2021, pp. 369–380.
https://doi.org/10.1061/9780784483374.037

[53] Touma, T., Daş, E., Burdick, J. W., Clark, E. B., Mackey, R., Feather, M. S., Fesq, L. M., Kolcio, K. O., and Prather, M., "Towards Robust, Resilient Ocean World Science Sampling Systems," *2023 IEEE Aerospace Conference*, Inst. of Electrical and Electronics Engineers, New York, 2023, pp. 1–9.
https://doi.org/10.1109/AERO55745.2023.10115718

[54] McMahon, J., Ahmed, N., Lahijanian, M., Amorese, P., Deka, T., Muvvala, K., Shakerin, K., Slack, T., and Wakayama, S., "REASON-RECOURSE Software for Science Operations of Autonomous Robotic Landers," *2023 IEEE Aerospace Conference*, Inst. of Electrical and Electronics Engineers, New York, 2023, pp. 1–11.
https://doi.org/10.1109/AERO55745.2023.10115533

[55] McMahon, J., Ahmed, N., Lahijanian, M., Amorese, P., Deka, T., Muvvala, K., Slack, T., and Wakayama, S., "Expert-Informed Autonomous Science Planning for In-Situ Observations and Discoveries," *2022 IEEE Aerospace Conference (AERO)*, Inst. of Electrical and Electronics Engineers, New York, 2022, pp. 1–11.
https://doi.org/10.1109/AERO53065.2022.9843445

[56] Marchetti, Y., Wagner, C., Twu, P., Cameron, M., Reeves, G., Tan-Wang, G., Chien, S., Hurst, K., Castano, R., and Wagstaff, K., "In-Situ Science Data Analysis and Curation for the Europa Lander Mission Concept," *Planetary Science, Informatics and Data Analytics Conference*, European Space Agency (ESA), 2022, https://ai.jpl.nasa.gov/public/documents/papers/Marchetti-PSIDAC-2022.pdf.

[57] Deutsch, L. P.*GZIP File Format Specification Version 4.3*, RFC Editor, Association Management Solutions, LLC (AMS), Fremont, CA, 1996, pp. 1–12, https://www.rfc-editor.org/info/rfc1952.

[58] Sterritt, R., and Hinchey, M., "Engineering Ultimate Self-Protection in Autonomic Agents for Space Exploration Missions," *12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'05)*, Inst. of Electrical and Electronics Engineers, New York, 2005, pp. 506–511.
https://doi.org/10.1109/ECBS.2005.36

[59] Truszkowski, W., Hallock, H. L., Rouff, C., Karlin, J., Rash, J., Hinchey, M., and Sterritt, R., *Autonomic Systems*, Springer London, London, 2010, pp. 173–186.
https://doi.org/10.1007/b105417_8

[60] Horn, P. J., "Autonomic Computing: IBM's Perspective on the State of Information Technology," IBM, 2001, https://api.semanticscholar.org/CorpusID:59693806.

[61] "OR-Tools," Google, 2022, https://developers.google.com/optimization [retrieved 27 Oct. 2022].

[62] Chien, S., "Formal Methods for Trusted Space Autonomy, Boon or Bane?" *NASA Formal Methods Symposium*, 2022, https://ai.jpl.nasa.gov/public/documents/papers/chien_nfm_2022.pdf.

[63] Wang, D., Russino, J. A., Basich, C., and Chien, S., "Using Flexible Execution, Replanning, and Model Parameter Updates to Address Environmental Uncertainty for a Planetary Lander," *Workshop on Integrated Execution (IntEx)/Goal Reasoning (GR), International Conference on Automated Planning and Scheduling (ICAPS IntEx/GP 2020)*, Oct. 2020, pp. 34–43, https://ai.jpl.nasa.gov/public/documents/papers/europa-lander-icaps2020-workshop.pdf.

[64] de la Croix, J.-P., and Lim, G., "Event-Driven Modeling and Execution of Robotic Activities and Contingencies in the Europa Lander Mission Concept Using BPMN," Jet Propulsion Lab., NASA JPL-CL-20-4302, 2020, https://dataverse.jpl.nasa.gov/dataset.xhtml?persistentId=hdl:2014/53293.

[65] Völzer, H., "An Overview of BPMN 2.0 and Its Potential Use," *Business Process Model and Notation*, Springer, Berlin, 2010.

[66] Bowkett, J., Kim, D. I., Nash, J., Moreno, D. P., Gildner, M., Thakker, R., Wehage, K., Kim, S.-K., Brinkman, A., Emanuel, B., Edlund, J., Ridge, B., Jain, A., and Backes, P., "Demonstration of Autonomous Sampling Techniques in an Icy Moon Terrestrial Analog," *2023 IEEE Aerospace Conference*, Inst. of Electrical and Electronics Engineers, New York, 2023, pp. 1–15.
https://doi.org/10.1109/AERO55745.2023.10115687


E. Atkins
*Associate Editor*