# Outcome Prediction and Explainability for Mission Operations of Autonomous Spacecraft

**Alberto Candela, Tiago S. Vaquero, Bennett Huffman, Nihal Dhamani, Federico Rossi, Rebecca Castaño**
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive
Pasadena, California 91109 USA
{Alberto.Candela.Garza, Tiago.Stegun.Vaquero, Bennett.W.Huffman, Nihal.N.Dhamani, Federico.Rossi, Rebecca.Castano}@jpl.nasa.gov

**Abstract**

As planning and autonomy in general become increasingly deployed on board spacecraft, missions will face a paradigm shift in how ground operations teams command and interact with the spacecraft: moving from specifying timed sequences of commands to high-level goals that on-board autonomy will elaborate based on the spacecraft's state and sensed environment. It will become increasingly difficult for operators to predict a mission's outcome as autonomous spacecraft venture into deeper space, react to unknown conditions, and face stronger communication constraints. However, data from simulated autonomous missions can be analyzed and leveraged, allowing operators to make informed decisions when selecting mission parameters and high-level goals. To this end, our paper presents a framework that gains insights from simulation data in order to help operators of autonomous spacecraft missions to predict, explain, and search for specific outcomes given a set of high-level goals. We show and discuss how our approach can help operators to better understand predictions, explore options and make informed decisions. Specifically, we describe a case study that builds upon previous work on simulated autonomous spacecraft missions to the Neptune-Triton system where the spacecraft uses an automated planning and execution framework to make onboard decisions (Castano et al. 2022; Vaquero et al. 2022).

## Introduction

Future space exploration missions will have advanced onboard autonomy capabilities to increase science return, improve spacecraft reliability, reduce operations costs, or even achieve goals that cannot be attained through regular ground operations due to communication constraints or limited lifetime. Examples of autonomy capabilities being developed for future mission include autonomous planning, scheduling and execution (Chi et al. 2019; Troesch et al. 2020), autonomous selection of scientific targets (Francis et al. 2017), autonomous fault management (Kolcio, Fesq, and Mackey 2017) and onboard data summarization and compression (Doran et al. 2020). Autonomy has already significantly increased the capabilities of Mars rover missions, enabling them to perform tasks such as autonomous long-distance navigation and autonomous data collection of new science targets (Estlin et al. 2012). Automated ground-based planning and scheduling, in particular, has been deployed on daily ground operations for the Perseverance rover (Yelamanchili et al. 2021) and is projected to be deployed on board in the near future (Rabideau et al. 2020).

As spacecraft become more autonomous, mission outcomes will be increasingly difficult to foresee, especially for missions that face very limited communications in environments with large uncertainties (e.g., deep space). There will be a need to build user trust in the decision making of the onboard planner. The Crosscheck system developed for the Perseverance rover (Yelamanchili et al. 2021) is an explanation tool that serves as an example. Moreover, the graphical user interfaces developed for the ASPEN-RSS scheduler (Chien et al. 2021) on the Rosetta Orbiter mission also has explanation features, in this case providing feedback on which constraints are preventing an observation from being scheduled.

While it is not possible to anticipate all potential scenarios that the spacecraft will encounter, the uncertainty related to the environment (e.g. likelihood that a plume will be active a certain latitude and longitude) and the spacecraft itself (e.g. the likelihood of components failures, variations in the duration of on-board activities, etc) can be modelled on the ground. A team can evaluate and simulate how likely the goals will be achieved given such models, what is the impact on the mission, and how that translates to progress towards the campaigns and science objectives. Users can inspect each subset of outcomes to analyze possible performance. However, navigating through the possible outcomes is not trivial as there may be many environmental and spacecraft variables involved in the analysis. Furthermore, mission simulations can take a long time to run, especially if they have a high fidelity.

One valuable notion is that data patterns from previous simulations can be identified and leveraged during the process of navigating through possible scenarios. To this end, our paper presents a framework that gains insights from simulation data in order to help operators of autonomous spacecraft missions to predict, explain, and search for specific outcomes given a set of high-level goals. We show and discuss how our approach can help operators to better understand predictions, explore options and make informed decisions.

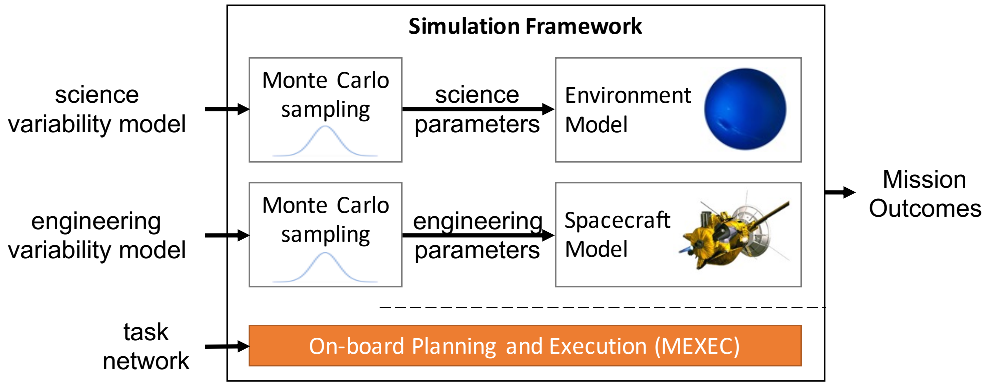



Figure 1: Simulation framework for autonomous spacecraft missions developed by (Castano et al. 2022; Vaquero et al. 2022). It uses a Monte Carlo simulation approach to generate and explore different scenarios. In summary, it consists of: a) an environment model that is defined by a set of science parameters, b) a spacecraft model that is defined by a set of engineering parameters, and c) an on-board planning and execution system (MEXEC) that works using task networks. All of these simulation inputs produce a set of mission outcomes.

To this end, we describe a case study that builds upon previous work on simulated autonomous spacecraft missions that use an automated planning and execution framework to make onboard decisions (Castano et al. 2022; Vaquero et al. 2022).

## Background

This paper builds upon previous work regarding operations for autonomous spacecraft (Castano et al. 2022; Vaquero et al. 2022). Specifically, we focus on simulated autonomous spacecraft missions to the Neptune-Triton system, which is an especially interesting setting because the significant light-speed latency, low available bandwidth, short duration of flybys, and dynamic scientific phenomena make autonomy highly attractive to fulfill primary mission objectives, but also make operations of such a mission very challenging.

The framework for these simulated missions uses MEXEC (Multi-mission EXECutive) (Troesch et al. 2020) as the core planning and execution system onboard the spacecraft. MEXEC is a flight-proven system that has been demonstrated on the ASTERIA CubeSat and was used on the JPL's Europa Lander Surface Mission Autonomy project (Wang et al. 2022). Also, MEXEC shares core reasoning components with the planning system used in the Perseverance rover's operations. Campaign goals are represented in *task networks*, which is the foundation of timeline-based temporal planning and Hierarchical Task Network (HTN) planning. Task networks are meant to be used by operators, mission planners, engineers and autonomy experts to represent their intent as goals. Goals are expressed in the form of tasks, including their pre-, post- and maintenance conditions, impact/effect constraints, temporal and resource constraints, priority, as well as ordering constraints and how the tasks decompose into sub-tasks hierarchically.

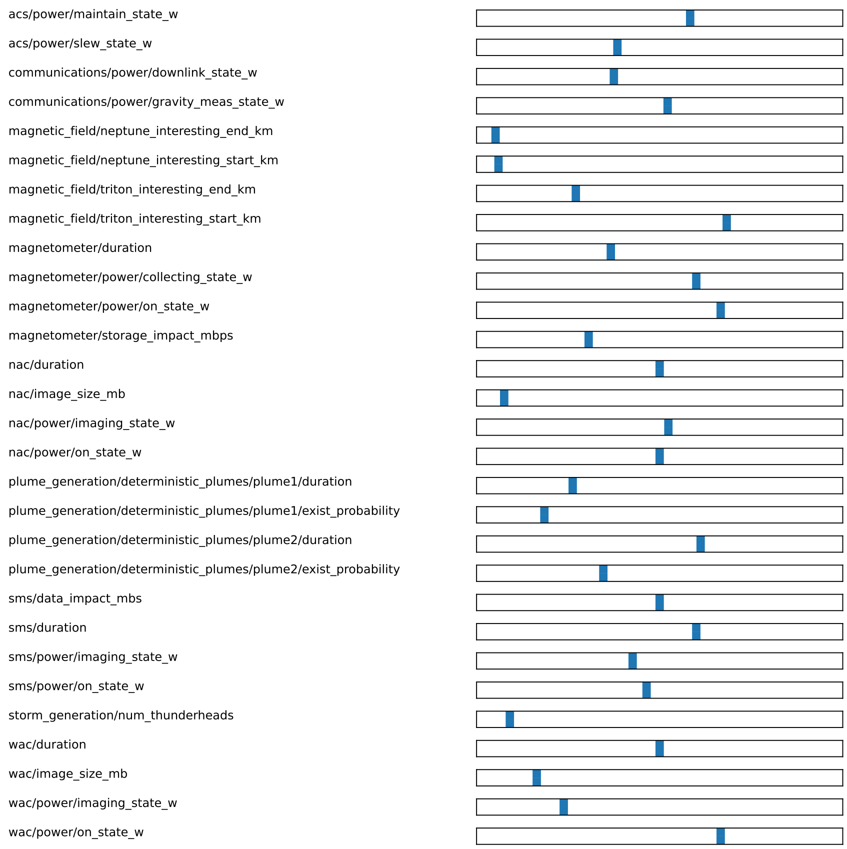
In such framework (Figure 1), variability models are first captured, i.e., users specify science and engineering parameters that can vary, and a model for that variability. For example, this can entail modeling activity duration uncertainty as a Pert or Gaussian distribution, or modeling activity effects probabilistically, or modeling off-nominal behavior of instruments and components, or modeling uncertainty of science phenomena models. Once variability information is captured, a Monte Carlo simulation approach is used for exploring different scenarios and conditions and investigating possible outcomes given a particular task network. The Monte Carlo simulation also requires a model of both the spacecraft and environment, this is what the performance of the automated planning and scheduling systems and the goals are tested against. The framework integrates both the Monte Carlo system and the simulator, and stores data from each simulation run into a database.

In total, previous work simulated 165 variations of the mission to the Neptune-Triton system. There was runtime replanning in these simulation runs. Two different task networks were used, each corresponding to a different orbit. Out of the 165 simulations, 90 were run using the first task network while 75 used the second task network. Specifically, the simulations had 29 science (uncontrollable) and engineering (controllable) input parameters. During each simulation, 8 outcome metrics or outputs (also known as key performance indicators (KPIs)) were evaluated. Figure 2 shows these mission variables in more detail.

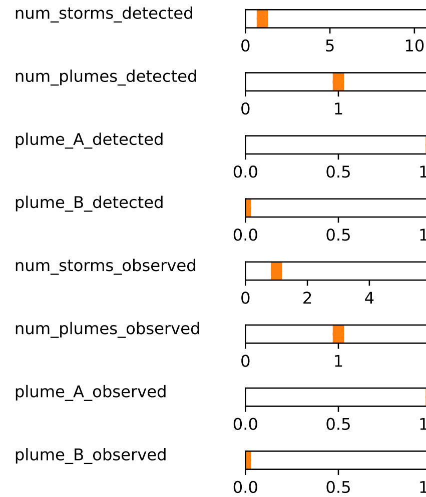
## Outcome Prediction Framework

This paper presents a framework that allows operators to predict, explain, and search for outcomes of autonomous spacecraft missions by leveraging data patterns from simulations. We present and develop a case study using the data that was collected using the simulation environment that was described in the previous section (Castano et al. 2022; Vaquero et al. 2022).

In order to offer the operations team a more complete overview of the potential behaviors of the on-board autonomy, we predict the various outcomes that may result from a given task network and a set of environment and spacecraft parameters by running an array of high-fidelity simulations.




(a) There are 29 simulation parameters (inputs) related to the science and engineering elements of the mission.




(b) There are 8 outcome metrics (outputs) that evaluate how many storms and plumes were detected and observed during the mission.

Figure 2: Simulation environment variables of the Neptune-Triton mission.

Our framework tends to favor Bayesian formalism for data analysis and prediction as it provides several benefits (Murphy 2013). First, Bayesian formalism provides mathematical rigor together with statistical and uncertainty guarantees, which are vital for safe spacecraft operations. Second, Bayesian formalism can improve explainability and transparency in the overall process as opposed to many black-box approaches for data analysis. Finally, Bayesian formalism aligns well with the aforementioned Monte Carlo-based simulation framework for the spacecraft missions.

Our framework consists of the following three elements:

1. **Prediction**: we start with a tool for fast and accurate outcome predictions given a particular task network and a set of specific mission parameters.
2. **Explanation**: we then describe a tool for conducting an exploratory analysis of the mission data in order to organize and explain different outcome distributions.
3. **Search**: finally, we describe an approach to direct the search for a specific outcome within the simulation environment.

The rest of the paper describes these three elements in more detail.

## Outcome Prediction

As we had discussed, data patterns from previous mission simulations can be inferred and leveraged. To this end, herein we employ Gaussian process regression (GPR). GPR regression has been extensively studied in the machine learning and statistics literature (Rasmussen and Williams 2006). It is Bayesian, data-driven approach to modeling complex relationships between input variables and output variables. GPR assumes that the function values are normally distributed with a mean and covariance matrix that are defined by a kernel function that measures the similarity between the input points.

The prediction of a new output variable given new input values is obtained by conditioning the GPR on the observed data. This results in a posterior distribution over functions, which can be used to obtain the predictive mean $\mu$ and variance $\sigma^2$ at any new input value. The mean function describes the expected value of the prediction, while the variance function describes the uncertainty of the prediction. As more data becomes available, the predictions tend to improve while the uncertainty is reduced. Figures 3 and 4 show examples of these notions on our dataset, specifically for the 90 simulation runs that correspond to the first orbit and task network.

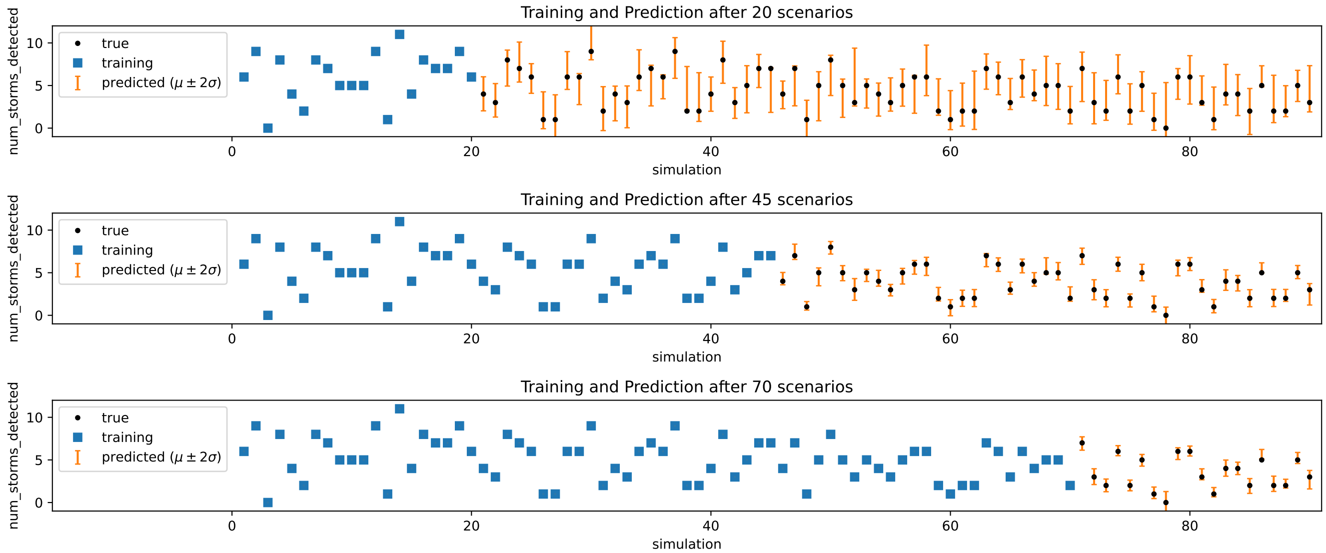



Figure 3: GPR predictions for the number of detected storms during a mission scenario. GPR predictions are probabilistic and described by a mean and a standard deviation ($\mu \pm \sigma$). At first, GPR predictions are less accurate and have a higher uncertainty. With more data, the GPR gets better at modeling the relationships between the mission parameters and the outcome metrics.

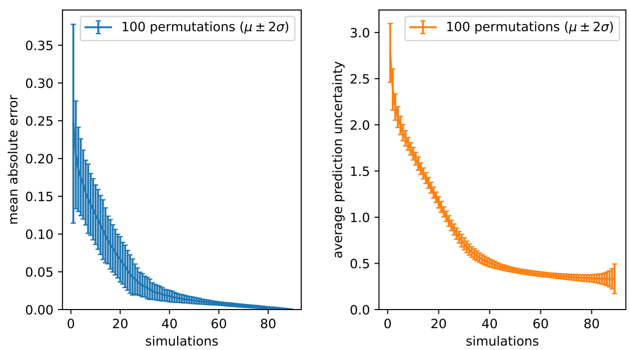



Figure 4: GPR prediction error (left) and average uncertainty (right) for the number of detected storms during a mission scenario. These plots show trends for 100 permutations of the dataset. As more data becomes available, both the error and uncertainty tend to decrease. For this particular example, the knee or elbow of the curve is at around 30 simulations, indicating that most of the data patterns have already been inferred by then.

Specifically, we applied GPR using the squared exponential kernel, which is a common function for modeling similarities and differences between data points. All the input and output data was preprocessed to achieve good results; it was rescaled between 0 and 1 using min-max normalization. Figure 5 includes examples of the training data together with small perturbation predictions for a couple of variables. GPR is able to learn how some mission outcomes are strongly coupled to certain input parameters while others are essentially uncorrelated.

## Outcome Explanation

Predicting outcomes is a useful step, but operators also need to understand these predictions. In other words, they need to be able to extract and visualize patterns in the simulation data in order to make informed decisions when planning a mission. To this end, our framework relies on machine learning for organizing and grouping the data.

In machine learning, clustering involves grouping together data points or objects based on their similarities or patterns in the data (Bishop 2007), without the need for predefined labels or categories. The goal of clustering is to identify natural structures or clusters in the data that can help to segment the data, gain insights, and facilitate decision-making processes.

In the machine learning literature, there are many clustering algorithms, each with its own benefits and drawbacks. Herein we employ Gaussian Mixture Models (GMM), a popular probabilistic clustering algorithm that assumes that the data is generated from a mixture of Gaussian distributions, each of which represents a cluster or a subpopulation in the data. GMMs assign each data point to one of the clusters based on the likelihood of belonging to each cluster. GMMs estimate the parameters of these underlying Gaussian distributions, namely the mean and covariance; hence, directly providing useful statistics for each group.

GMMs require users to specify in advance the number of clusters in the model. One common approach to selecting the appropriate number of Gaussian components for a GMM is the Bayesian Information Criterion (BIC). The BIC is a measure of model complexity that balances the goodness-of-fit of the model to the data and the number of parameters used in the model. The BIC penalizes models with more parameters, which reduces the risk of overfitting and helps

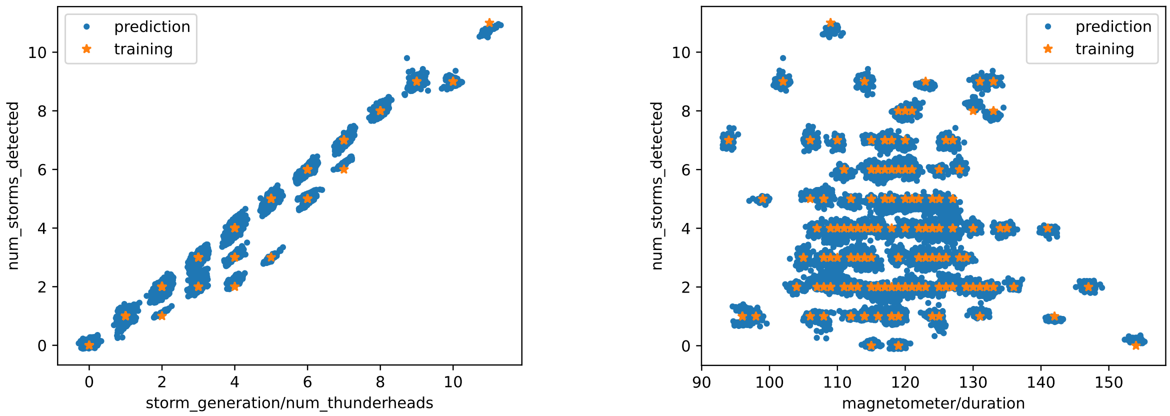



Figure 5: Mission outcome prediction using GPR. For mission parameters that are similar to the training data, the model is consistently predicting similar outcomes. The model is able to learn the patterns where some mission outcomes are strongly coupled to certain input parameters (left), whereas others are essentially uncorrelated (right).

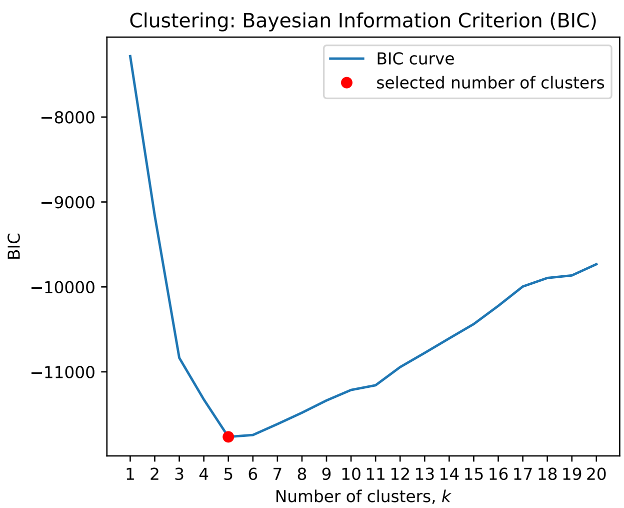



Figure 6: Application of the BIC approach for estimating the best number of clusters $k$. For our dataset, 5 clusters provide the best solution as they minimize the BIC value.

to identify the optimal number of clusters that best explain the data. Lower BIC values indicate a better model fit with fewer parameters, and the optimal number of clusters can be determined by selecting the model with the lowest BIC value. In our study, five clusters better fit the data according the BIC method (Figure 6). After performing clustering, our tool shows operators the statistics for each cluster in terms of the variability parameters and their corresponding outcomes (Figure 7).

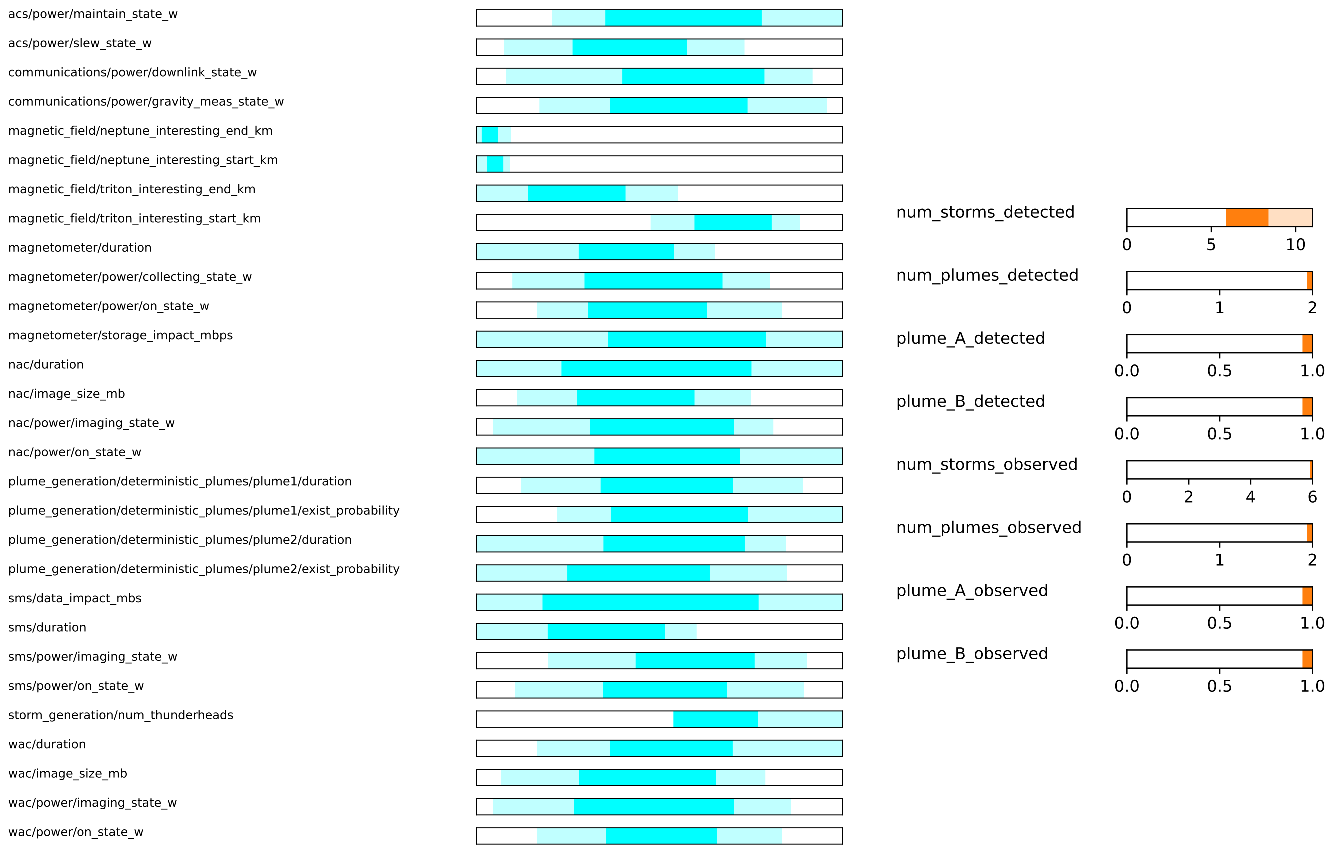
The clusters are hard to visualize and compare to each other since there are 29 different variability parameters (inputs) and 8 outcomes (outputs). This work explores two popular techniques used for visualizing high-dimensional data: Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbor Embedding (t-SNE).

PCA is a statistical method that reduces the dimensions of a dataset while retaining the most important information in the data (Jolliffe 2002). PCA works by projecting high-dimensional data onto a lower-dimensional subspace while preserving the maximum amount of variation in the data. The new dimensions in the lower-dimensional subspace are called principal components. These principal components are orthogonal to each other and can be ranked based on the amount of variation they explain in the data.

On the other hand, t-SNE is an algorithm that is specifically designed for visualizing high-dimensional data in a low-dimensional space (van der Maaten and Hinton 2008). Unlike PCA, t-SNE does not rely on linear transformations of the data. Instead, it converts the high-dimensional similarity between datapoints into a probability distribution and then tries to optimize a low-dimensional similarity distribution that is as close as possible to the high-dimensional one.

Figure 8 shows 2D representations, using both PCA and t-SNE, of the 5 clusters that were previously derived. These visualizations allow us to see which clusters are more similar to each other, and vice versa. In this case, both PCA and t-SNE show how clusters 0, 3, and 4 are more closely related to each other, while clusters 1 and 2 are the most dissimilar.

Our framework also makes the clustering results more explainable and understandable for operators, especially for complex models with many variables, as the one in our case study. For this we rely on SHapley Additive exPlanations (SHAP) values, which are a method for explaining the output of machine learning models (Lundberg and Lee 2017). SHAP values assign a numerical value to each feature in a data point, indicating how much that feature contributes to the predicted outcome of the model. When it comes to clustering, SHAP values can be used to explain which input parameters have the greatest impact on each cluster. As an example, Figure 9 shows the SHAP values for cluster #4, which consists of scenarios where a large number of storms were detected. These values show that the input parameter with the greatest impact is indeed the number of storm thun-




(a) Variability parameter (input) statistics.

(b) Outcome (output) statistics.

Figure 7: Clustering of mission scenarios and outcomes. This example shows the statistics of cluster #4, which represents top-performing scenarios, especially those where a high number of storms were detected. Saturated colors represent one standard deviation regions, whereas light colors indicate min-max ranges.

derheads that were generated for the simulation, which is reasonable and confirms the relationship that the GPR predictor learned (Figure 5).

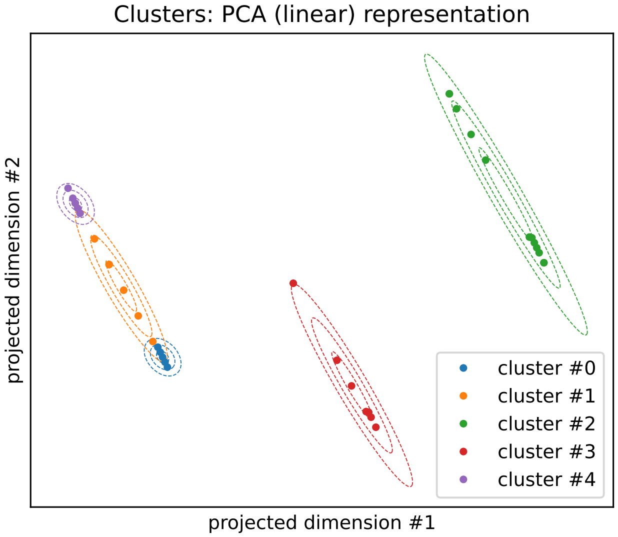
## Outcome Search

Once we have a prediction tool and a better understanding of different possible outcomes, we proceed to search for input parameters that produce a desired outcome. In our study, the search process focuses on controllable engineering parameters and not on uncontrollable science inputs. For simplicity, our search process is conducted on each outcome metric independently from the others.

Our search process employs Bayesian optimization. It is a model-based approach for navigating through black-box functions that are expensive to evaluate (Snoek, Larochelle, and Adams 2012), which makes it appealing for our purposes. It works by constructing a probabilistic model of the objective function, called a surrogate, which is used to guide the selection of the next point to evaluate (in our case a scenario to simulate). This guidance process is based on the values that are predicted by the surrogate together with their uncertainties, ultimately balancing exploration and exploitation so as to reduce the number of function queries. The surrogate model is updated with each new evaluation, incorporating the information gained from previous evaluations to improve its predictions.

GPR is a popular choice for constructing the surrogate model in Bayesian optimization. The key idea behind using GPR in Bayesian optimization is to use the surrogate model to estimate the acquisition function, which is a heuristic that balances exploration and exploitation. The acquisition function is used to select the next point to evaluate; here we employ the lower confidence bound (LCB) criterion, which combines predicted values together with their uncertainties. By using the acquisition function, the algorithm can efficiently search the input space and converge to a global solution with just a small number of expensive evaluations.

Bayesian optimization with GPR has been successfully applied to a wide range of applications, including hyperparameter tuning of machine learning models, design optimization of physical systems, and drug discovery (Shahriari et al. 2016). It is a transparent, powerful, and flexible optimization method that can handle noisy and high-dimensional functions, and can be customized to suit dif-



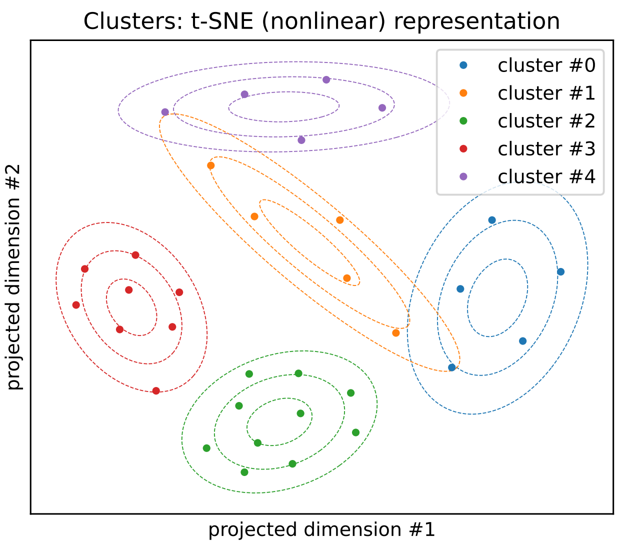




Figure 8: 2D visualizations of the similarities and differences among data clusters. These visualizations employ linear and nonlinear methods: PCA (top) and t-SNE (bottom), respectively. Both approaches show that out of the five clusters in the dataset, clusters 0 (blue), 1 (orange), and 4 (purple) are more closely related or each other, while clusters 2 (green) and 3 (red) are the most dissimilar.

ferent problem domains.

Figure 10 illustrates how our outcome search works. At first, little information is known from only a few random scenarios that are not likely to produce the desired outcome; this represents a blind search process. The GPR surrogate model is updated with this data and the LCB acquisition function suggests which scenario should be simulated next. The approach tends to favor exploration first, that is, simulating diverse scenarios. After running some simulations, there is more information, hence the search process converges and now focuses on exploitation, that is, on simulating similar scenarios that achieve a specific outcome.

The Bayesian optimization approach can be complemented with our GPR predictor. Since running simulations is expensive and slow, fast GPR predictions can serve as "true" outcomes until the process converges to a solution that can be simulated afterwards, further reducing the number of simulation runs. The GPR predictor can also be used to test and finetune the Bayesian optimization process before running any actual simulations.

## Conclusions and Future Work

This paper describes a framework for autonomous spacecraft missions that can help operators to predict, explain, and search for outcomes given a set of high-level goals (i.e., a task network). We present and develop a case study that builds upon previous work on simulated autonomous spacecraft missions to the Neptune-Triton system where the spacecraft uses an automated planning and execution framework to make onboard decisions (Castano et al. 2022; Vaquero et al. 2022).

The GPR predictor learns a model of the simulation environment that can generate much faster results, which is especially useful for exploratory, testing and prototyping purposes. Even if the GPR predictor starts by not being as accurate as a high-fidelity simulation, it can be further improved as more simulation data becomes available for training.

Using GMMs to cluster outcomes based on data patterns and similarities is useful as it provides cluster statistics right away; this can help identify off-nominal scenarios. Linear and nonlinear methods of data visualization can be used to intuitively show users which clusters are most similar to each other, and vice versa. Furthermore, SHAP values tell users which input parameters have the greatest impact on each cluster.

Outcome search using Bayesian optimization provides a transparent strategy for directing the process of navigating through desired outcomes. It starts by exploring diverse scenarios to get a better sense of the underlying distribution of scenarios and outcomes. When more data is available, it focuses on generating specific scenarios that produce a desired outcome. This search process can be fine-tuned and/or accelerated with the GPR predictor so as to reduce the number of expensive simulation runs.

We have shown how our approach can be used for mission operations of autonomous spacecraft but we want to emphasize that these methods are also applicable to other missions that use automated planning and execution. For example, the Europa Lander Surface Mission Autonomy project also uses MEXEC. Automated planning and scheduling will be deployed on board the Perseverance rover in the near future, and these tools could help a lot on this endeavor.

Our work can be used on the ground for design purposes, and ultimately we would like to explore how it could be used in an online fashion to interact with the planner and produce better plans, either during the initial plan creation, or interactively during execution to refine the plan as uncertainty and outcomes become known.

Future work will continue to improve this outcome prediction framework. For instance, it will directly incorporate

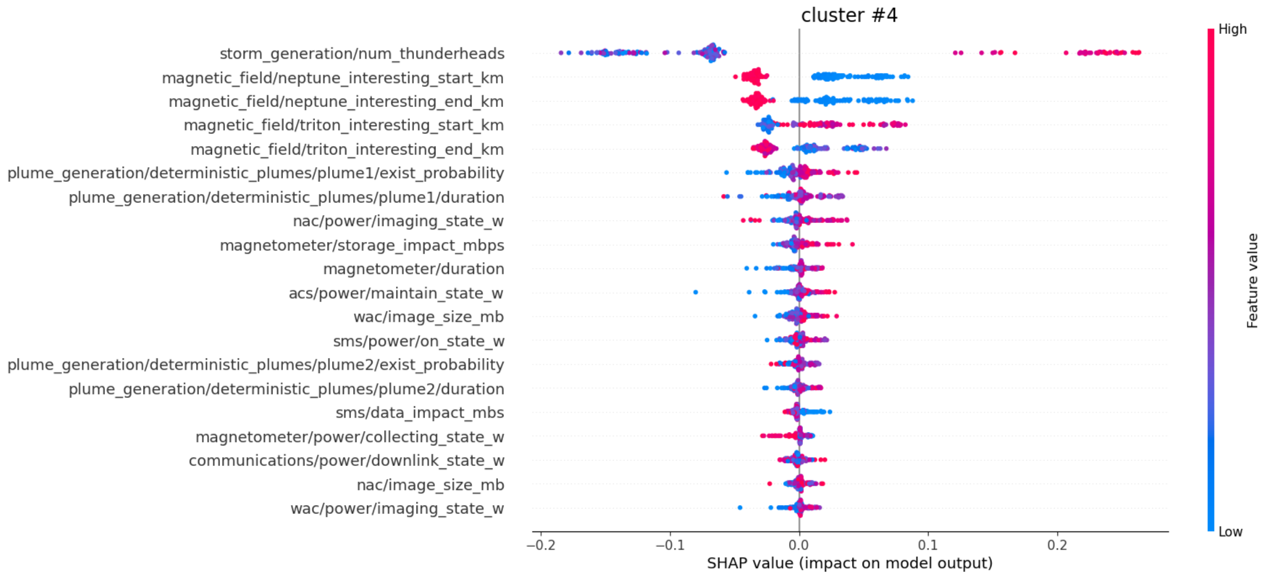



Figure 9: Clustering explanations using SHAP values. The most important features are ranked, as well as their impact on a particular cluster. In this case, the number of storm thunderheads that were generated for the simulation is the input parameter with the greatest impact on cluster # 4, which comprises missions where a high number of storms were detected.

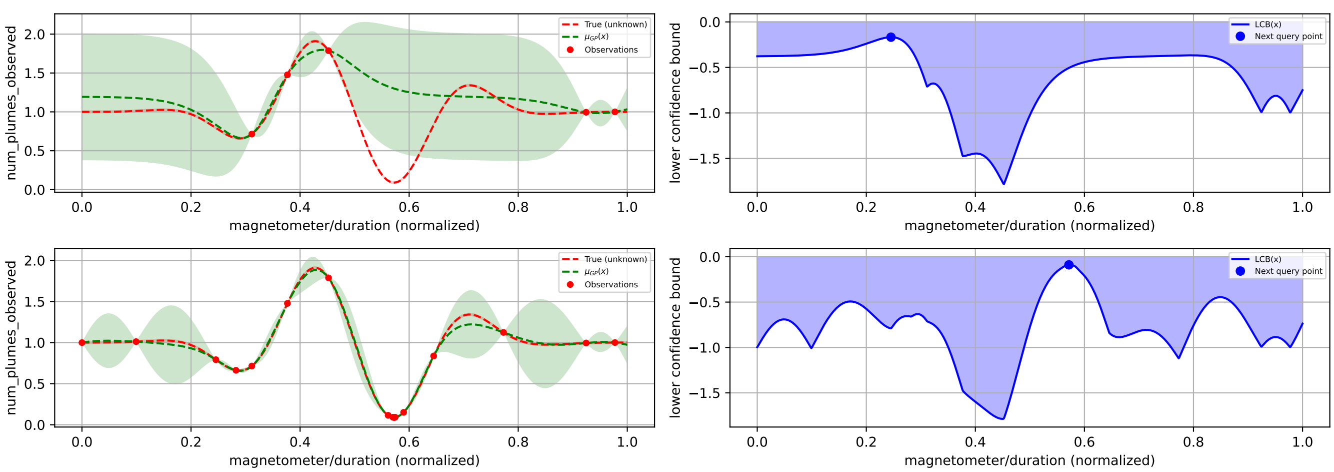



Figure 10: Outcome search process using Bayesian optimization. This particular example illustrates the process of finding underperforming mission scenarios in terms of the number of observed plumes. For simplicity and visualization purposes, just one input parameter (magnetometer/duration) is shown. At first (top row), the search focuses on exploration as little information is known from only a few random scenarios. The GPR surrogate model is updated with this data (left column) and the LCB acquisition function (right column) suggests which scenario should be simulated next based on predicted values together with their uncertainties. After running more simulations (bottom row), the search process now performs exploitation since there is more information about which scenarios produce the desired outcome.

task network parameters such as task durations and constraints. We also want to expand the outcome search approach so it supports multi-objective criteria and acceleration via distributed computing. Finally, future work will investigate how to adapt and apply these tools to other JPL missions that rely on automated planning and execution. We are especially interested in applying this method to the CADRE and Mars 2020 missions.

## Acknowledgments


The research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (NASA) (80NM0018D0004).

## References


Bishop, C. M. 2007. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 1 edition. ISBN 0387310738.

Castano, R.; Vaquero, T.; Rossi, F.; Verma, V.; Van Wyk, E.; Allard, D.; Huffmann, B.; Murphy, E. M.; Dhamani, N.; Hewitt, R. A.; Davidoff, S.; Amini, R.; Barrett, A.; Castillo-Rogez, J.; Choukroun, M.; Dadaian, A.; Francis, R.; Gorr, B.; Hofstadter, M.; Ingham, M.; Sorice, C.; and Tierney, I. 2022. Operations for Autonomous Spacecraft. In *2022 IEEE Aerospace Conference (AERO)*, 1–20.

Chi, W.; Agrawal, J.; Chien, S.; Fosse, E.; and Guduri, U. 2019. Optimizing Parameters for Uncertain Execution and Rescheduling Robustness. *Proceedings of the International Conference on Automated Planning and Scheduling*, 29(1): 501–509.

Chien, S. A.; Rabideau, G.; Tran, D. Q.; Troesch, M.; Nespoli, F.; Ayucar, M. P.; Sitja, M. C.; Vallat, C.; Geiger, B.; Vallejo, F.; Andres, R.; Altobelli, N.; and Kueppers, M. 2021. Activity-Based Scheduling of Science Campaigns for the Rosetta Orbiter. *Journal of Aerospace Information Systems*, 18(10): 711–727.

Doran, G.; Lu, S.; Liukis, M.; Mandrake, L.; Rebbapragada, U.; Wagstaff, K. L.; Young, J.; Langert, E.; Braunegg, A.; Horton, P.; Jeong, D.; and Trockman, A. 2020. COSMIC: Content-based Onboard Summarization to Monitor Infrequent Change. In *2020 IEEE Aerospace Conference*, 1–12.

Estlin, T. A.; Bornstein, B. J.; Gaines, D. M.; Anderson, R. C.; Thompson, D. R.; Burl, M.; Castaño, R.; and Judd, M. 2012. AEGIS Automated Science Targeting for the MER Opportunity Rover. *ACM Trans. Intell. Syst. Technol.*, 3(3).

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. 2017. AEGIS autonomous targeting for ChemCam on Mars Science Laboratory: Deployment and results of initial science team use. *Science Robotics*, 2(7): eaan4582.

Jolliffe, I. 2002. *Principal Component Analysis (2nd ed)*. Springer.

Kolcio, K.; Fesq, L.; and Mackey, R. 2017. Model-based approach to rover health assessment for increased productivity. In *2017 IEEE Aerospace Conference*, 1–13.

Lundberg, S. M.; and Lee, S.-I. 2017. A Unified Approach to Interpreting Model Predictions. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Murphy, K. P. 2013. *Machine learning : a probabilistic perspective*. Cambridge, Mass. [u.a.]: MIT Press. ISBN 9780262018029 0262018020.

Rabideau, G.; Wong, V.; Gaines, D.; Agrawal, J.; Chien, S.; Kuhn, S.; Fosse, E.; and Biehl, J. 2020. Onboard Automated Scheduling for the Mars 2020 Rover. In *Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation for Space*, i-SAIRAS'2020. Noordwijk, NL: European Space Agency.

Rasmussen, C. E.; and Williams, C. K. I. 2006. *Gaussian processes for machine learning.* Adaptive computation and machine learning. MIT Press. ISBN 026218253X.

Shahriari, B.; Swersky, K.; Wang, Z.; Adams, R. P.; and de Freitas, N. 2016. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proceedings of the IEEE*, 104(1): 148–175.

Snoek, J.; Larochelle, H.; and Adams, R. P. 2012. Practical Bayesian Optimization of Machine Learning Algorithms. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'12, 2951–2959. Red Hook, NY, USA: Curran Associates Inc.

Troesch, M.; Mirza, F.; Hughes, K.; Rothstein-Dowden, A.; Bocchino, R.; Donner, A.; Feather, M.; Smith, B.; Fesq, L.; Barker, B.; and Campuzano, B. 2020. MEXEC: An Onboard Integrated Planning and Execution Approach for Spacecraft Commanding. In *Workshop on Integrated Execution (IntEx) / Goal Reasoning (GR), International Conference on Automated Planning and Scheduling (ICAPS IntEx/GP 2020)*. Also presented at International Symposium on Artificial Intelligence, Robotics, and Automation for Space (i-SAIRAS 2020) and appears as an abstract.

van der Maaten, L.; and Hinton, G. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(86): 2579–2605.

Vaquero, T. S.; Rossi, F.; Castano, R.; Jasour, A.; van Wyk, E.; Dhamani, N.; Barrett, A.; Huffman, B.; and Jorritsma, M. 2022. A Knowledge Engineering Framework for Mission Operations of Increasingly Autonomous Spacecraft. In *Workshop on Knowledge Engineering for Planning and scheduling (KEPS), International Conference on Automated Planning and Scheduling (ICAPS)*.

Wang, D.; Russino, J. A.; Basich, C.; and Chien, S. 2022. Analyzing the Efficacy of Flexible Execution, Replanning, and Plan Optimization for a Planetary Lander. *Proceedings of the International Conference on Automated Planning and Scheduling*, 32(1): 518–526.

Yelamanchili, A.; Agrawal, J.; Chien, S.; Biehl, J.; Connell, A.; Guduri, U.; Hazelrig, J.; Ip, I.; Maxwell, K.; Steadman, K.; and Towey, S. 2021. Ground-based Automated Scheduling for Operations of the Mars 2020 Rover Mission. In *Proceedings Space Operations 2021*.