SSC23-WII-01

Developing Intelligent Space Systems: A Survey and Rubric for Future Missions

Mary Dahl, Christine Page, Kerri Cahoy
Massachusetts Institute of Technology
70 Massachusetts Ave, Cambridge, MA 02139
marydahl@mit.edu

Evana Gizzi
NASA Goddard Space Flight Center
8800 Greenbelt Rd, Greenbelt, MD 20771
evana.gizzi@nasa.gov

ABSTRACT

Space exploration continues to inspire the development of advanced technologies to explore the universe. From moon landings and the Mars rovers to the recent successful deployment of the James Webb Telescope, space exploration continues to push the limits of what is possible in both science and technology and to pave new ways for the discovery of our galaxy. While we have seen great advancements and barriers broken, we remain limited in our ability to optimize science discovery without onboard intelligent capabilities to enable complex system missions. In this paper, we focus on AI technologies and cross-disciplinary directions for sparking research and development for space applications. This paper attempts to bridge two disparate fields of research, to enhance the development of intelligent space systems by providing a comprehensive survey of existing technologies and showcasing a strategic rubric for future advancements.

1 Introduction

Space exploration has inspired innovation in performing remote tasks effectively and consistently. Continued improvement of autonomous technology will advance our capabilities to collect data in Earth orbit, navigate interplanetary environments, and bring complex instruments further into the solar system than ever before. We are limited in our ability to optimize science discovery without onboard intelligent capabilities and distributed systems. Artificial intelligence (AI) provides rapid and adaptive problem-solving, although researchers do not yet fully understand the capabilities and applications for AI. The space industry has recognized this need for many years. In a 2006 article by Chien et al.,[1] AI was identified to be useful for detecting events of scientific interest, for automating planning, and for decision making and task execution. In a 2007 survey, Girimonte and Izzo[2] identified AI as useful in space applications for distributed system computation, situational self-awareness, and for multidisciplinary design optimization. A 2010 survey by Frost[3] focused on the necessary capabilities for widespread application of AI in space. Frost found that future systems must be able to operate for the long-term, operate reliably, guarantee success, and run concurrently with other systems.

Although we have made progress since these surveys, we remain limited in our application of AI for space in several key areas. Taking a multidisciplinary approach allows for non-AI engineers and scientists to better understand how these tools can be leveraged to improve their mission capabilities. In this work, we present a survey of AI usage in the space industry and a rubric for assessing future AI capabilities. We refer to the Johns Hopkins Applied Physics Lab's (APL) AI Technology 2020 Roadmap,[4] which presents multi-term goals and key investment areas for their organization. Space research is broad, so we apply the following limitations in this work. First, we focus on successfully deployed or flight-ready missions, with some exceptions for novel research. Second, we do not cover AI applications for post-downlink processing. Third, we do not include robotics, rovers, or autonomy with humans in the loop. Lastly, we focus our survey on works published in 2017 or later.

Our paper begins by categorizing AI approaches that have been used on space missions, highlighting key features of the aerospace problems which make the AI approaches viable solutions. We define critical for AI implementation: **(1) mission operational planning** (including activity planning, downlink prioritization, and navigation), **(2) satellite data processing** (including intelligent instruments, on board science missions, and distributed spacecraft mission (DSM) communications), and **(3) spacecraft health management** (including fault detection and smart data retrieval. For each category, we analyze example missions and novel research. We note where AI is not yet being used, and provide recommendations for tools and methods in AI which would be well-suited for these areas. We conclude by reviewing the implications of our findings by developing a rubric for assessing the need for AI in a particular system.

2 Early Applications of AI in Space

The first use of artificial intelligence for a space application was in 1990, when "Spike" was developed to serve as the scheduling system for the Hubble Space Telescope.[5] Spike received desired observations from astronomers around the world. Based on the pointing, power, thermal, communications, and exposure constraints for each observation, Spike generated a schedule using rule-based constraint algorithms, including procedural search, rule-based heuristic search, and even an artificial neural network (ANN). This neural net was different than those used today, but still had the capabilities of parallel computing to quickly schedule and reschedule as new requests came in.

The first use of AI onboard a spacecraft was Deep Space 1 (DS-1), launched in 1998. DS-1 was a satellite test bed for novel technologies, including autonomous navigation and remote intelligent self-repair. The navigation software, AutoNav,[6,7] later reused on the Deep Impact Spacecraft,[8] was used to estimate the location of DS-1 with an optical navigation algorithm. It then slewed the spacecraft and remained locked on its target. DS-1 also performed the Remote Agent Experiment,[9] which was the first time a spacecraft's flight software was AI-controlled.

In 2000, Earth Observing 1 (EO-1) spacecraft was launched with the Autonomous Sciencecraft Experiment (ASE) on board.[10] ASE was designed to show capabilities of onboard science analysis and mission planning, using the Continuous Activity Scheduling Planning Execution and Replanning (CASPER) system. EO-1 analyzed images for features of scientific interest, such as thermal anomalies, clouds, and floods, by using support vector machines with offline supervised learning.[11]

These missions demonstrate three capabilities for AI in space: planning missions and activities, navigation, and image processing.

3 Artificial Intelligence Categorization

Artificial intelligence is a discipline containing a multitude of models, applications, and paradigms. The relationship between many of them can be seen in Figure 1. Not all of these disciplines will be discussed in this survey; we focus applications of machine learning, constraint satisfaction, and evolutionary algorithms. Each of these types are defined and compared in detail in Figure 2. In this section, we review the sub-areas of AI research which underpin the surveyed works discussed in Section 4.

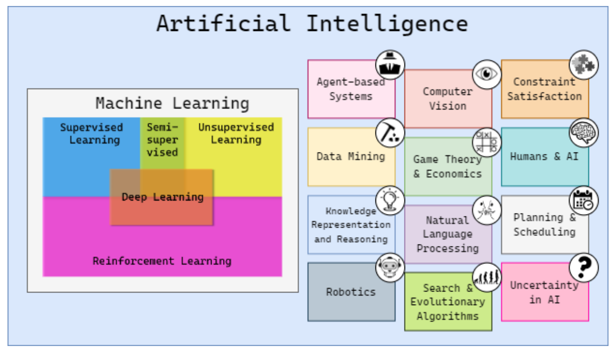

Figure 1: The relationship between different types of AI. The AI umbrella contains many different disciplines, but this paper focuses only on ML, constraint satisfaction, and evolutionary algorithms.

3.1 Machine Learning

Machine learning (ML) refers to any type of AI that focuses on the development of algorithms that function autonomously by learning from existing data. ML models discover inherent patterns in the data which is used to train the model. There are four main methods for training: supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. Additionally, it should be noted that deep learning is a distinct subcategory of machine learning which can employ any of the mentioned forms of training shown in Figure 1.

3.1.1 Supervised Learning

Supervised learning is a form of ML where the model is trained on *labeled* datasets. The label provides the correct output for each input. For instance,

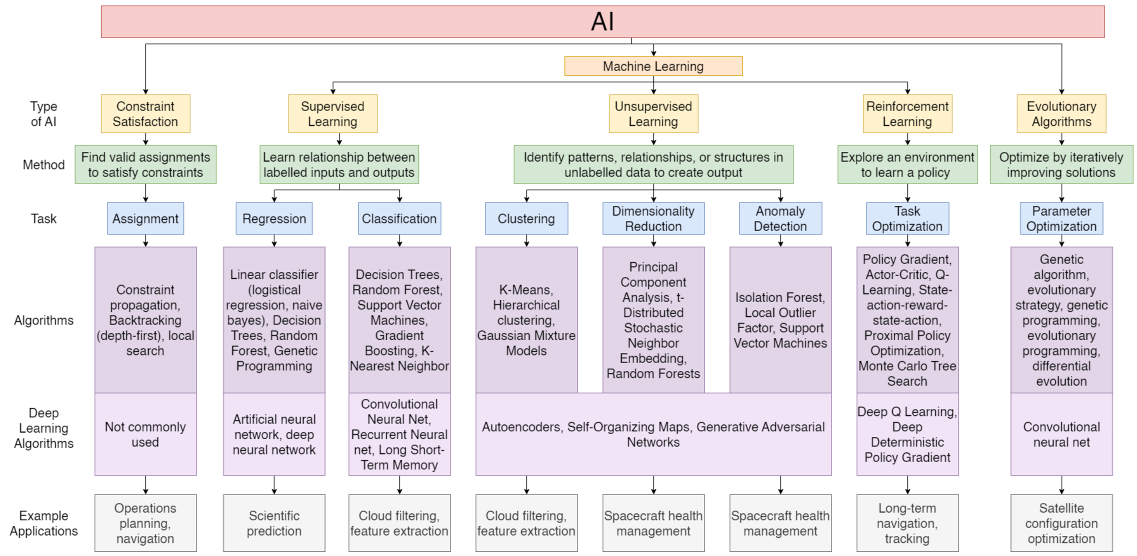

Figure 2: A summary of the types of AI discussed in this paper. This tree diagram shows the types of data and strategies that would be employed for utilizing each method, as well as examples for the applications that are useful for each.

an input could be an image of the handwritten letter "A" and the corresponding desired output would be the selection of the letter "A". During training, the model learns the relationship between the features of the input (ex. handwritten text) and the corresponding output (ex. typed text) in order to predict outputs for unseen input data (ex. handwriting samples). The supervised learning algorithm works best when there is a clear cause-effect relationship between the input features and their outputs, when the relationship does not change significantly over time, and when there is sufficient labeled data to capture the underlying patterns in the data. Common drawbacks to supervised learning techniques include needing a large quantity of labeled data, which can be time-consuming to generate, as well as needing a long time to train. One example of supervised learning for space applications is classifying unknown astronomical objects by training on preexisting labeled data from observational surveys.

3.1.2 Unsupervised Learning

Unsupervised learning, meanwhile, uses *unlabeled* training data. These algorithms find inherent patterns, relationships, or structures within the data itself. For instance, if an algorithm is given pictures of dogs and cats, the AI can begin to distinguish differences in their features in order to group them. It is best suited to learn from the patterns or overall structure of the data itself, rather than focusing on target outcomes. Unsupervised learning can be used for grouping similar instances together (clustering), simplifying input data while preserving its structure (dimensionality reduction), and identifying unusual information (anomaly detection). Drawbacks to unsupervised learning include unpredictable outputs, as the model may find patterns unintended by the user, leading to inaccurate conclusions. It can also be difficult or impossible to understand the way that the model has grouped things together to correct this. An example space application is taking large amounts of unlabeled planetary surface images and processing them to categorize features such as craters by the pattern of their size, shape, and shading.

3.1.3 Semi-supervised Learning

Semi-supervised learning is, as the name implies, a combination of supervised and unsupervised learning techniques. It labels only a portion of the training data to get the model "on the right track" similar to a supervised learning model, while also not needing a large, labeled dataset like in unsupervised learning. In space, this can be used for spacecraft health monitoring and anomaly detection.

3.1.4 Reinforcement Learning

Reinforcement learning is used for determining the optimal way to perform in an environment (or, policy for behavior in an environment). An agent is set up with potential states it can be in, actions it can take, and rewards it can receive for performing certain actions or reaching certain states. The agent is trained to maximize the total reward it can receive over time. Reinforcement learning is used where decision-making is sequential or a mission is long-term, such as navigating a rover on the surface of Mars or optimizing the target pointing and tracking strategy of a space telescope.

3.1.5 Deep Learning

Deep learning utilizes an artificial neural network (ANN) and can be applied to supervised learning, unsupervised learning, or reinforcement learning. ANNs are computation techniques designed to mimic the brain by using layers of nodes (or neurons) that transmit signals. These neurons each have their own weights that they assign to data being passed through them, and the training process involves optimizing the values of these weights. Deep learning is able to learn from large amounts of data and can automatically identify features from raw data for ML tasks such as image recognition. For example, it can be used for celestial object identification, satellite image analysis, and onboard data processing.

3.2 Constraint Satisfaction

Constraint satisfaction is an AI technique to find valid assignments for variables to satisfy a set of constraints. Each variable is given a domain, or a set of assignments it can take, and the problem is given in a form of constraints for those variable assignments. It creates plans by exploring this model and finding a route to bring the system into a valid configuration. This type of AI does not require the planner to have knowledge of the underlying system, as it has entirely been abstracted. The most commonly used constraint-based tool is the Planning Domain Definition Language (PDDL),[12] although many variants on this have been created. This type of AI is best suited for scheduling and planning operations for a spacecraft or rover, or optimizing possible configurations of space exploration missions.

3.3 Evolutionary Algorithms

Evolutionary algorithms (EAs) are optimization algorithms that mimic natural selection and evolution in order to iteratively optimize a design space. They take a population of of potential solutions, such as values for a control algorithm, and perform bio-inspired operators such as mutation, crossover (recombination), and selection to evolve the assignments towards better solutions. Evolutionary algorithms are robust and highly adaptable, capable of solving complex problems. They are particularly useful when the search space is large, complex, or poorly understood, such as determining optimal configurations for satellite constellations or scheduling the actions of these satellites to maximize coverage and minimize cost.

4 Survey of Intelligent Space Systems

In the following section, we provide a comprehensive background of surveyed works which employ AI for space missions.

4.1 Mission Operations Planning

We define mission operations planning as any AI that is developed to manage the operations of a spacecraft, including planning the day-to-day activities, determining the order information will be transmitted to the ground or to other satellites, and navigating the spacecraft from one location to another. Mission operations is one of the most active areas of AI research for several reasons. For one, automation at a certain level for these tasks is necessary. Beyond communication with spacecraft not being instantaneous, the geometry of orbits and space travel also means that communication blackouts are inevitable. Instantaneous and persistent communication with a spacecraft is impossible, so certain activities must be automated. That level of automation can be increased to optimize the quantity and quality of tasks being performed by the spacecraft on its own. Through this section, we examine how each of these actions have been performed historically and currently, and identify areas for improvement.

4.1.1 Activity Planning

Activity planning, also called scheduling, is the process by which the actions of a spacecraft are decided. These actions can include taking scientific measurements, performing maneuvers, transmitting data, and charging batteries. It will usually correspond to a mode change for the on board computer. Key to activity planning is knowledge of the constraints on the spacecraft. These constraints can be geographical or temporal (measuring the correct target), mechanical (power management, pointing, keep out windows), electrical (data storage), thermal (hardware overheating), and others. One of the earliest examples of AI mission design and planning is the activity planner on EO-1,[10] CASPER. CASPER used a constraint-based planning algorithm with a technique called iterative repair to rapidly change plans to accommodate constraints from the spacecraft and the scientific needs of the mission. When an assignment is in conflict, iterative repair randomly chooses a variable and changes the value to minimize conflicts. It continues doing this until a solution is found, or it times out. CASPER also had a ground based counterpart, called ASPEN. CASPER was improved and used in the Intelligent Payload Experiment (IPEX) CubeSat mission.[13] The improved CASPER was robust to more on-board constraints, such as unexpected file sizes and power drain. Although unrelated to mission planning, it would be remiss not to mention that IPEX was also the first mission to train a machine learning algorithm on orbit, performing image classification to identify clear images, the edge of the Earth, clouds, and outer space.

These early examples of constraint-based planning are limited by the feasibility of enumerating all possible constraints. As missions become more complex, with more subsystems and capabilities, the number of constraints increase significantly. These planners are also limited by their onboard processing speed for searching the state space.

There are more advanced versions of planning languages and architectures. For instance, the NASA Platform for Autonomous Systems (NPAS)[14] turns away from "brute-force autonomy," where all possible constraints and edge cases must be hard-coded, as this leads to some constraints inevitably being missed. Instead, NPAS focuses on "thinking-autonomy," where live models are generated and general strategies are programmed on cause-effect trees. The system can then autonomously plan and solve issues. NPAS is focused on not only solving autonomous problems but providing information on the condition of the system by the cause-effect trees.

Eventually, however, a growing state space requires either more powerful and/or distributed hardware or only estimating solutions. One algorithm designed for low-power hardware is the planner for the Mars 2020 rover, done with a Monte Carlo simulation.[15] In this method, a greedy algorithm is first used to plan activities according to priority, without heed to all constraints and without backtracking. The simulation then finds problematic activities, adjusts, and continues until all constraints are satisfied.

Such an estimation-based scheduler can also use metamodeling, or making models of commonly known models. Algorithms such as radial basis functions, Kriging, and support vector regression all efficiently center in on a "close enough" optimal value. This was used for finding satellite-to-satellite communication windows.[16] Evolutionary algorithms can be used for similar effect. Genetic algorithms have been used for the motion design of a space tether for a payload orbital transfer to increase efficiency, and reliability, and reduce uncertainty related to human-interference.[17]

4.1.2 Downlink Prioritization

A subset of activity planning is downlink prioritization. Spacecrafts generate enormous quantities of data and only a portion can be sent back to Earth. All spacecraft need to deal with communication window limitations, and solar system missions need to contend with constraints on use of the Deep Space Network. There can also be constraints on the power of the hardware (radio or optical).

AI can be used to decide which data to send down. Some data can be quickly determined to not contain scientific information, such as images that are covered in clouds. AI can be used to immediately discard these measurements. This is discussed in more detail in Section 4.2.2.

Another problem that faces missions with infrequent downlink periods and long data collecting periods is onboard data storage. Data is stored in onboard buffers, and data is lost if the spacecraft attempts to put more data onto a full buffer. Assigning priority to data transfers can be done to maximize the buffer space for additional data collection. An algorithmic approach to this problem can apply a heuristic that manages priority assignments.[18]

4.1.3 Navigation

Autonomous navigation capabilities allow for more complex mission architectures in more distant or remote locations. The current standard for AI enabled spacecraft navigation on spacecraft is AutoNav,[7] the navigation software used on DS-1. The optical navigation system estimates the spacecraft's current position with images taken of asteroids, a triangulation algorithm, and a least-squares regression. It then autonomously maneuvers with a linear controller and keeps visual lock on the target with an estimation algorithm. This type of AI is affected by how well the current position estimate is made, limiting the performance based on how well triangulation is performed.

Improvement on AutoNav can be found in multiple references. In a survey by Izzo et al,[19] different algorithms are described with their capabilities for navigation. Evolutionary algorithms are used to develop the Global Trajectory Optimisation Problem

(GTOP) database,[20] a database of pre-solved navigation problems. Tree searches are used for navigation problems with small domains, as they can converge to an optimal solution. Machine learning, specifically deep learning, is potentially powerful for this application, but a lack of training databases makes it difficult to implement on a wide scale.

An interesting case of applying AI techniques to missions comes from the Mars Express (MEX) spacecraft.[21] Arriving at Mars in December 2003, its gyroscopes approached their end of life in 2017, normally indicating the end of the mission. Instead, however, new software was sent to the spacecraft to perform gyroless flight by utilizing a supervised learning random forest algorithm, trained on previous flight data. The algorithm both learned the differences between the flight with and without gyroscopes and was able to generate commands to operate MEX. However, this solution encountered a similar issue to the one that Izzo predicted: insufficient training data for long time horizon prediction.

A method for accessing more training data for supervised learning algorithms is by algorithmically generating it. By utilizing optimal control theory, samples of realistic trajectories that include past, current, and future states can be generated arbitrarily. This has been done for moon landing by modeling the dynamic environment between the moon and a lander while adding a factor of randomness to avoid overfitting the ML model.[22] It has also been done for electric propulsion systems for low thrust target trajectory design.[23] For the latter, a simple feedforward neural net was trained on the generated data and resulted in a controller that is comparable to one designed by humans.

A more advanced version of the visual navigation system on AutoNav can be found on HERA, an ESA mission to investigate the aftermath of the asteroid system impacted by DART. Because HERA needs to get into close proximity with the asteroid, precise autonomous navigation is needed. Rather than triangulating its location, HERA uses centroiding-based navigation to track the asteroid target. When doing low altitude flybys, HERA uses feature tracking, either only with its visual camera, or by utilizing sensor fusion with its visual camera, thermal images, and an altimeter, during lower flybys.[24] HERA shows the progress made since AutoNav, utilizing more sensors and allowing for more precise and complex maneuvers.

The most important lesson from navigation is that, even if a system is well suited for AI implementation, limitations may prevent its inclusion. Supervised techniques need training data. Unsupervised techniques bring a level of uncertainty that may be considered too risky for certain missions. Solutions for these problems can commonly be seen in robotics research, with control barrier functions, or Lyapunov functions which show mathematical proof of safety and stability.[25] Increased application of safe boundary functions can improve confidence in AI systems.

4.2 Satellite Data Processing

For science based missions, optimizing payload performance is integral to success. Traditionally, data collection is simple: the instrument is programmed to turn on at a certain time and make measurements. When the spacecraft comes into range of a terminal for the communications system it is utilizing, either a ground station or the Deep Space Network, the satellite either automatically or is commanded to send data down. This flow is shown in Figure 3, along with potential AI options to improve it.

These AI improvements need to be made as new instruments (hyperspectral imagery, synthetic aperture radar, etc) generate more data. A recent survey[26] identified the most important applications for satellite data processing as image recognition, object detection, and change detection. Deep learning and neural nets are among the most promising methods for performing these tasks. We categorize these capabilities broadly as "on board data processing," focusing on collecting better data, sending less data down, and prioritizing data downlink. Each of these has different ways that AI can be used.

4.2.1 Intelligent Instruments

To collect better data, we can improve how we choose data to collect. Intelligent pointing is a strategy where a spacecraft autonomously determines where it should point its scientific instruments to collect the highest quantity or the most salient data. Most research thus far has been done for the mobile phone industry. One example is using a self-learning framework to track mobile stations and terminals.[27] First, sensor fusion is used to estimate the location of the antenna on the ground station, then, a support vector machine is used to point, and, finally, tracking is done with deep reinforcement learning to maximize the signal strength. Similar pointing work can be done with a recurrent neural net and deep Q learning, taking advantage of the fact there is only one factor to optimize (the mobile signal), which is independent of the control factors (the antenna pointing).[28]

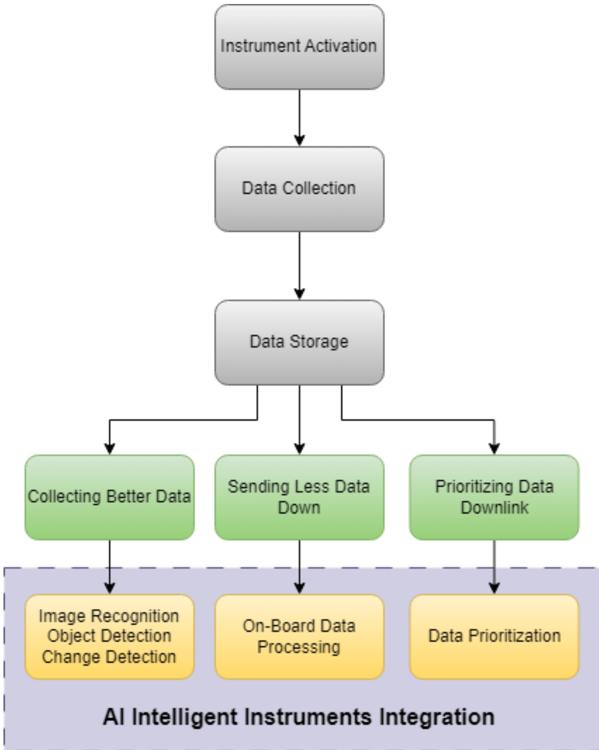

Figure 3: The flow of utilizing artificial intelligence to improve data processing on satellites. Three examples paths that can lead to potential improvement: better data collection, better data processing, and downlink prioritization.

This type of work of maximizing the desired signal is not limited to mobile phones. For example, on GOSAT-2, a satellite designed for greenhouse gas monitoring, the TANSO instrument has intelligent pointing capabilities to identify cloud-free locations by applying a mask to its field of view. It then shifts where it is pointing to focus on those spots.[29]

Because many points of interest to satellites are independent of the pointing commands, deep learning may be a powerful tool to continue to improve this work. A monitoring algorithm can search for objects of scientific interest in the field of view of the instrument. When one is found, the quaternion that corresponds to that pointing vector can be stored and commanded to the ADCS system. Alternatively, multiple fields of view on instruments could be utilized. An imaging system that has a wide field of view camera can be used to scan an area for objects of interest. When one is identified, a camera with a narrow field of view can examine it in detail. Systems like this would allow for the search and discovery of unknown phenomena, like natural disasters.

4.2.2 On Board Data Processing

There are two forms of on board data processing to consider: analyzing data for scientific viability and performing scientific calculations on board. Both of these decrease the amount of information needing to be downlinked. Both strategies have merit, and different science mission examples will be analyzed. In particular, OPS-SAT[30] provides a compelling case study for the types of on board processing of interest to the aerospace engineering community.

OPS-SAT is a 3U CubeSat platform developed by the European Space Agency (ESA) to allow individuals to run AI experiments on an orbital platform with on board hardware. OPS-SAT provides access to TensorFlow Lite, and is the first platform to deploy and use it on orbit. It also provides open source machine learning algorithms, where users can upload training data to create their own model. OPS-SAT has several apps users can utilize, including an app with a CNN called SmartCam that classifies data from an onboard optical sensor as "Earth," "Edge of Space," or "Bad,"[31] and an app called Software as a Service for Machine Learning (SaaSyML) that makes it easier for users to access training data and a wide variety of pre-implemented supervised algorithms, including classifiers, clusters, outliers, and regression.[32]

In 2022, the ESA reported[33] that there were twenty-one publications generated out of OPS-SAT experiments, with six of them involving on board AI experiments: implementation of a Spiking Neural Net,[34] algorithm comparison for cloud detection,[35] augmenting digital signal processing with a CNN,[36] comparing an ANN on different hardware for cloud detection,[37] fault detection with a RNN and long short-term memory,[38] and comparing CNNs on FPGAs and CPUs.[39] There are additionally twelve more experiments that do not have publications yet, but have been approved for deployment or have already been deployed.[40] Four of them are new AI-based experiments: a machine learning based super-resolution algorithm, a deep neural net for image processing, a deep Q network for intelligent pointing, and a RNN for intelligent pointing.

Examining these provides a compelling case study for the types of on board data processing missions that are of interest to the aerospace community. Of these AI-enabled missions, six are primarily for image processing and/or algorithm comparison, one is signal processing, one is fault detection, and three are attitude determination and control and/or intelligent pointing.

There are other orbital applications for which AI is relevant. These include fast ocean front detection utilizing deep learning models,[41] weather analysis utilizing supervised ML techniques including SVM, long short-term memory, and decision trees,[42] rainfall estimation using microwave radiometer data utilizing an ANN,[43] and estimating wind speed utilizing an ANN.[44] One application that showed up multiple times is cloud detection. Cloud detection is a highly desirable capability for on-orbit applications, as, if you can determine that an image you took is mostly clouds, it does not need to be downlinked. In a survey on cloud detection, utilizing papers from 2004-2018, 28 different machine learning techniques in 6 categories (SVM, fusing convolutional features, deep learning, decision trees, ANN, and probabilistic semantic analysis) were identified from the literature, and their performance was compared to traditional techniques.[45]

There were several important conclusions from this survey. Performance was not necessarily correlated with the complexity of the algorithm, as the average performance of the traditional algorithms was 91.5% and the AI algorithms was 91% – an insignificant different, suggesting they are both valuable. AI algorithms have an advantage in certain difficult cases, such as identifying thin clouds and differentiating clouds for snow. This study did not consider runtime in their assessment, which can make a significant difference. A comparison between RF and U-Net, a type of CNN, on OPS-SAT[35] found that, while U-Net performed better (92% accuracy vs. 90%, improvement in snow cases, as well as being translationally equivariant), it also took much longer. Performance was comparable on other hardware,[46] but it introduces a key consideration in AI development for spacecraft: hardware requirements. Similar results were found in earlier implementations of cloud segmentation on OPS-SAT,[47] where multiple neural nets were tested, and the best results were found on a FCN, which was able to utilize the FPGA to full effect.

Although not a focus of this paper, small satellites have increasingly taken advantage of different System-on-a-Chip (SoC) devices, including CPUs, GPUs, FPGAs, DSPs, and combinations of these.[48] There are strategies that are worth considering when designing for small satellites, such as size weight and power (SWaP) and a reduction of system latency by reducing dependency on ground systems.[49] To this end, there have been custom computers made for small satellites.[50,51] An alternative to traditional SoCs is neuromorphic computing,[52] where elements of the computer system are modeled after the nervous system. Neuromorphic computing mimics the continuous variable weights in a neural network, so it's specifically designed for AI computing tasks and can be deployed on the edge. This technology has been developed by Intel and other companies,[53] and has been flown on TechEdSat-13,[54] showing successful operation of an AI/ML experiment for 225, or 2.5 orbits.

When developing onboard AI enabled processing, it is important to consider hardware at all stages of development; it is of little use to design a complex algorithm to save time on orbit when it takes longer to run than it would take to downlink the data. Second, all these applications focus on curating collected data for downlink rather than performing full science missions onboard. There is potential for continued improvements through examining this capability.

4.2.3 Distributed Spacecraft Mission Communications

Small satellite constellations have been proposed and, in some cases, deployed. Multiple, co-working satellites allow for more data to be collected and for distributed calculations to increase on orbit processing. In a 2020 survey, Curzi et al.[55] identified three problematic areas mega constellations contribute to: complex management, excessive communication, and space traffic. Traditional RF communications is infeasible to manage tens or hundreds of satellites, as their high frequency of ground station communication would crowd RF bands and make it difficult for new missions to be deployed. On board AI, in conjunction with crosslink communications, provides a method to consolidate this.

Two strategies for constellation data routing are opportunistic and scheduled. In opportunistic data routing, data is transmitted whenever pre-set conditions are met and a connection to a ground station or another satellite is available. This allows for dynamic adaptation to changing conditions in the constellation, but also tends to have data congestion during peak times and delayed or lost data if not coordinated properly. Scheduled data routing sets pre-determined transmission times to optimize the data routing throughout the constellation. This approach helps to manage network resources, providing a predictable data transmission pattern while also mitigating the risk of data congestion. However, this strategy is less flexible and may not fully utilize the network's capacity if the scheduled transmission times do not coincide with periods of optimal satellite connectivity.

Early concepts for automizing multi-agent communications simulated distributed systems to infer states of the satellite formation for task allocation.[56] Combinations of opportunistic and scheduled methods – scheduling with on board, rapid rescheduling when opportunities require them – are one of the more promising routes of research. A basis for this research can be found in non-AI solutions, such as a multi-integer linear programming solver, the Scheduling Planning Routing Inter-satellite Network Tool (SPRINT).[57] It combines a ground based planner with individual satellite planners to schedule, plan, and re-plan the constellation's activity under dynamic conditions. When a satellite is commanded to make an unplanned observation, it breaks down scheduled data routes, solves its own data volume allocation problem, and passes a new plan along the network. It increases data volume, decreases latency, and can be deployed onto the hardware of small satellites.[58] Scaling is an issue for MILP based solutions, and estimation methods based on AI may be necessary for continued improvements.

Evolutionary algorithms such as genetic algorithms, simulated annealing, and particle swarm optimization have been used effectively in distributed planning, as well as machine learning methods including deep reinforcement learning. A comparison of these based on data from a satellite operator found that deep reinforcement learning performed fastest with strong performance, while a particle swarm-genetic algorithm hybrid had lower power consumption and good robustness (while maintaining performance).[59]

Dynamic rescheduling is also well-suited to AI applications. Small spacecraft swarms could use a deep learning cognitive cooperative data scheduling protocol, powered by an RNN, to improve the efficiency of data collection and transmission. The proposed protocol uses deep learning techniques to adapt to the dynamic conditions of the swarm and improve the overall performance of the system.[60]

A crucial part in deploying AI in space applications is ensuring that the method is well-matched to the problem it is attempting to solve, as well as lightweight and modular enough to be deployed on hardware. Conventional CNNs can struggle with satellite scheduling problems, while graph neural nets, a specific type of neural net for working with graph data, provide a topography that matches the problem it is solving, seeing increased performance over traditional path-planning algorithms.[61]

4.3 Spacecraft Health Management

Spacecraft health management refers to any tools that are used to keep a space mission operating at its intended performance level. Autonomous systems should recognize anomalies, predict potential failures, and if necessary, perform corrective measures. The unique and often harsh environments on spacecraft need resilient solutions for ensuring their health. Fault detection systems are designed to identify the existence, the location, and the type of a fault. It is important to have high detection rates with low false alarm rates.

AI can enhance spacecraft health management by offering data-driven techniques for constant monitoring, fault prediction, fault detection, fault attribution, and system correction. Traditional methods of fault detection often rely on predetermined rules, models, and thresholds. These methods can be time-consuming, susceptible to errors, and don't adapt to changing conditions.[62] They also struggle when the spacecraft encounters an unknown anomaly, requiring human intervention to resolve the conflict.

Models that combine AI with probabilistic models provide solutions for these problems. Dynamic bayesian methods are able to reason about anomalies by modeling uncertain dynamic systems without heavy-duty simulations in scenarios with noisy data and partial observability.[63] As with all things, however, as the system becomes more complex and data volume increases, new methods are necessary. AI proves valuable for processing the amount of telemetry data that a satellite generates. This data can be used for a range of applications such as error detection, prediction, data summarizing, and visualization, all contributing to the efficient management of the spacecraft's health.[64] Neural nets can be utilized to manage a spacecraft's health status, both detecting and classifying anomalous behavior based on training data.[65]

Anomalous faults, or those that are not known to exist, and thus cannot be part of a supervised learning training set, can be managed in a few ways. With semi-supervised learning, the system can be trained on a small labeled dataset (normal and anomalous behavior) and a large unlabeled dataset. This training enables the system to detect anomalies even when the specific fault conditions were not part of the initial training set.[65] Alternatively, a combination of predictive and AI modeling can perform a system-level, application agnostic methods for fault diagnosis.[66]

On a longer mission scale, AI-enabled systems that use a neural network such as long short-term memory could continuously monitor the health status of various subsystems and automatically alert operators about potential issues, adjusting to new circumstances and recognizing patterns over time. They may also be able to handle complex time-series data.[67]

5 Artificial Intelligence Assessment, Needed Capabilities & Research Recommendations

From this research, we have created an AI Capability rubric that emphasizes the capabilities effective AI systems have or should have. This rubric, shown in 1, contains 13 criteria we have identified to consider for AI adoption in space systems. Basis for this rubric takes inspiration from the AIAA Intelligent Systems Technical Committee Roadmap for Intelligent Systems in Aerospace[68] and from the EASA Artificial Intelligence Roadmap.[69] By looking at and grading potential systems on this rubric, we can find new capabilities and research areas in intelligent space systems that show promise.

5.1 Analysis of the Rubric

In this section, we will discuss what each of the 13 criteria are and why they were chosen.

Improvement Over Current State-of-the-Art: Perhaps the most obvious criterion, but using AI systems to replace non-AI systems is almost always done to achieve technical improvement or to supply a capability that does not yet exist. Showing this improvement is a key aspect to any integration of AI, which requires experiments that put both systems in equivalent scenarios and compare them. Criteria for assessing AI systems are plentiful in the field and will not be covered in detail here, but three examples are accuracy, or the results being nearest to the optimal value, quality, or the lack of defect, and completeness, or providing a full solution to the problem.

Consistency of Output: Volatility is a common concern with AI systems, or the potential to supply different results when given similar inputs. This can happen when models are overfit to their training data, meaning they only know how to handle the specific examples they are initially taught and vary wildly when different from that. In space especially, it is important to know what a system may do, as systems are expensive and difficult to recover if they enter fault states or are damaged.

Availability of Training Data: For many systems, using AI may initially seem like a powerful tool, but it is limited in efficacy by a lack of pre-existing training data. Even if training data exists for certain problems, it may be overly specific for individual scenarios or may not have enough variety. There are techniques that can be used to generate additional data or modify what exists to cover additional cases.

Risk Benefit: Risk benefit is not simply "how bad could it be if this system fails," as different mission parameters have different risks. A mission that is designed to test and utilize AI would be reliant on the system, but it is well worth the risk of performing it, as it is the primary goal. For others missions, if AI is only being used to augment performing another task, it failing and putting that task at risk is a much greater cost.

Explainability: AI systems are often subject to volatile trust from users; the moment a poor decision is made, the less likely subsequent information will be believed. For informative systems, such as disaster prediction, it is important for the users working with the returned data to understand why the AI reached its final conclusion. With additional explanation, incorrect conclusions may be better understood and able to be corrected for.

Robustness Under Space Environment: While this often refers specifically to hardware, the radiation-hardness of the equipment and its ability to function under hazards, there is important consideration to the ability of the software to remain robust. While training, the AI needs to be able to set checkpoints it can retrieve progress from when facing radiation interrupts.

Robustness Under Resource Constraints: The AI needs to remain functional when in the "worst case" scenario for space status. The AI needs to be able to resume the work it was doing if the spacecraft needs to move to a low power mode or a low communication mode.

Necessity for Human Intervention: Spacecraft naturally will not have instantaneous communication available with human operators, due to communication black out windows because of orbital geometry, limited power for communications systems, or for scheduled time with the Deep Space Network for deep space missions. If a system needs to be reset or manually commanded whenever an unknown is encountered, it may not be well suited for space

Table 1: The rubric by which any considered AI system can be rated. When considering adding an AI system, each of these categories shall be given a score from 0 to 3, where 0 is the system is incapable of fulfilling that criteria, 1 is is fills it poorly, 2 is it fills it on par with state of the art, and 3 is it improves upon state of the art.

	Criteria	**Description**
1	Improvement Over Current State-of-the-Art	When given the same inputs, the solution returned by the system is an improvement on the results from the non-AI counterparts. This improvement includes accuracy, quality, and completeness.
2	Consistency of Output	When given the same inputs, the system gives a consistent, correct output. In similar scenarios to a known output, the system gives similar outputs. It does not produce a solution that is drastic or unexpected. Its outputs follow the required rules and requirements at all times.
3	Availability of Training Data	The data to train the agent is high quality, easy to access, plentiful, and represents a multiple of potential situations. If it is not currently available, it can easily be generated. It is free from bias that will skew the result.
4	Risk Benefit	If the system fails to produce an output or produces an off-nominal output, the mission will not be at risk of ending. This includes not causing mechanical or electronic damage to the spacecraft or the environment around it.
5	Explainability	The system is able to supply explanations to users as to how it reached its final conclusion. It does not need to explain the technical details, but rather the individual physical aspects that lead to its final decision. The explanation can easily be understood and is complete and logical.
6	Robustness Under Space Environment	The system can act as intended under abnormal or unknown inputs. For space scenarios, this may include radiation effects, equipment malfunction, and unexpected hazards. The system maintains its ability to function, even if it enters a "safe" mode.
7	Robustness Under Resource Constraints	The system can act as intended under expected space conditions. This includes limited power, communications black outs, low gravity, and radiation.
8	Necessity for Human Intervention	The system can act as intended under expected space conditions, where human interaction is limited. The system can act independently from human intervention. The system does not need to be constantly commanded to reset.
9	Computational Cost	The system does not incur significant power, time, or data requirement when compared to a non-AI counterpart. It is able to complete its tasks in the time required to perform the mission.
10	Operation and Integration Complexity	The system is well-explained and understood by those using it. It can be integrated onto spacecraft hardware without requiring expensive, custom hardware.
11	Maintainability	The system is able to be updated and maintained easily. The system does not require a complete overhaul in order to improve and alter features. The system can be easily restarted.
12	Provability	The system can mathematically be proven to not enter any unsafe conditions. This can be because of barrier functions or other security features.
13	Legality and Morality	The system does not break any laws, including those about AI implementation and outer space treaties. The system has been assessed regarding its ethical implications.

deployment.

Computational Cost: Because spacecraft are often limited on power, space, and computation, the amount of strain adding an AI system must be considered carefully. An AI system could show a small increase in accuracy but require so much more power that it cannot operate frequently enough to match performance of a less accurate system. This is especially important for CubeSat missions.

Operation and Integration Complexity: For AI systems that are designed to be generically compatible with multiple spacecraft missions, it is important that they can easily be operated and integrated by multiple users. If specific, complicated hardware is necessary, it may be difficult for integration on multiple missions.

Maintainability: Upgrading and updating software with new information is crucial to ensuring longevity of a mission. For segmentation algorithms, for instance, if it is being trained on the ground, uploading new weights based on collected data is important to see consistent improvement of the system.

Provability: The most effective way to ensure the safety of an AI system is to ensure that it cannot enter an unsafe state to begin with. Being able to prove that a system has boundaries that prevent unsafe states is not always possible, but it provides an extra layer of security.

Legality and Morality: Like all technology, the use of AI should not perpetuate harm. While less obvious for space applications, it still remains important to assess AI systems for their inherent biases and flaws.

5.2 Use of the Rubric

The intent of this rubric is to capture the qualities that a successful AI system should have. When assessing the development and integration of a new AI system, each of the 13 criteria should be considered. If the system is incapable of filling that criteria, it gets a score of 0. If it does it poorly, it gets 1, if it does it equally as well as a non-AI counterpart, it gets 2, and if it does it better, it gets a 3.

An optional rule for using the rubric is weights. After scoring each criteria, weights, decimal values ranging from 0-1, can be applied to emphasize certain capabilities more than others. For instance, if the AI does not require training data, criteria 3, "Availability of Training Data," can be given a weight of 0. If a system is very sensitive to any risk to the mission life, criteria 4, "Risk Benefit," can be given a weight of 1. Weights should only be applied with significant consideration, as they could be used to bias or manipulate results. The purpose of using a rubric is not to create a scenario where a proposed system gets a high score: it is to critically and honestly *assess* the potential of that system. A system scoring low on one aspect of the rubric is not necessarily a no-go sign for that system: it, instead, points to an area for research improvement to support the development of that system.

5.3 Example Rubric Usage

One application identified in this survey was intelligent instrumentation. Traditional sweeps of sensors to gather data rely either on pre-existing knowledge of target locations or on sufficient ground coverage to ensure the target is captured. A spacecraft could instead analyze samples it measures, identify objects of interest based on prior training or anomaly detection, and command the instrument to point at that target. One example of this technology would be for a camera on a disaster-monitoring satellite. The satellite could identify forming storms, wildfires, or other natural disasters and track them immediately, rather than waiting for human operators to identify the images on the ground. This system would be trained on a dataset of natural disasters in different locations, states of formation, and orientations. We now apply the rubric to this technology.

1. Improvement Over Current State-of-the-Art: Comparing the capabilities of this satellite to a single satellite without this pointing technology, this technology is an improvement. It allows for disasters to be tracked immediately, providing key, early information. **Score: 3**

2. Consistency of Output: This criteria depends on the implementation, however, providing it with a complete training set should provide consistent identification of targets. The pointing algorithm's track-to-target is the larger concern with consistency, ensuring that it always takes the most efficient route. **Score: 3**

3. Availability of Training Data: Environmental disasters are commonly studied with remote sensing, so there is a large quantity of data available to identify anomalies. **Score: 3**

4. Risk Benefit: If this system did not function properly, the satellite would revert to the comparison case, where it cannot directly identify

natural disasters and take more specific images, but can still take images on a schedule. **Score: 2**

5. Explainability: By utilizing a supervised learning algorithm to identify targets, understanding the agent's reasoning in picking targets should be possible. **Score: 3**

6. Robustness Under Space Environment: This system suffers no additional constraints than the state-of-the-art. **Score: 2**

7. Robustness Under Resource Constraints: Because this system focuses on target prioritization, once it identifies a target to track, it doesn't need to constantly be taking pictures of locations that are not of interest, This suggests a better resource usage than state-of-the-art. **Score: 3**

8. Necessity for Human Intervention: Nothing on this system would require humans to reset the functionality. The system is not being re-trained on orbit, so there is no chance for faulty data to enter the training set. The pointing system malfunctioning would be on par with state-of-the-art. Because the system removes the need for operator commands, it overall performs better. **Score: 3**

9. Computational Cost: This technology requires two separate algorithms to be run on images with a set sampling frequency. Depending on how fine that frequency is, this could have large computational strain on the system to achieve desired performance. **Score: 1**

10. Operation and Integration Complexity: The satellite would already need pointing capabilities for the communications and power systems, so there would be no additional complexity from the state-of-the-art to integrate this system. **Score: 2**

11. Maintainability: As a satellite, it is difficult to maintain the systems other than commanding for bus resets. This is no worse than state-of-the-art, however. **Score: 2**

12. Provability: The pointing system can be proven to be safe by employing barrier functions. **Score: 3**

13. Legality and Morality: There is potential for this technology to be utilized for espionage and tracking, however, being on a satellite, the pointing slew rate and camera ground sample distance required would be greater than the satellite could likely provide. **Score: 3**

Total Score: 31/39 = 80%

This system scores highly on the rubric, suggesting it is a strong candidate for further research. As it is only in the preliminary phases, it is expected that certain scores will go up and down as the technology matures. It is important to continue to assess this score as development continues, and, should it ever become unacceptably low, to reassess the development of the technology.

6 Conclusion

We have analyzed the use of AI in the space sector and elucidated its benefits and detriments for different mission types. While surveying the current state-of-the-art, we have found places where AI should be continued to be implemented but, perhaps more importantly, we have found limitations and considerations necessary before using AI on any system. This paper has proved that intelligent systems can improve spacecraft performance only when carefully designed to match the system needs that they are trying to fill.

Too often, the pace of research and development in the field of AI often exceeds our ability to keep up. When the large language model ChatGPT and its visual language model companion Visual GPT released, there was a massive influx in research interest. Already, it is garnering interest in how it can be utilized for space applications. Currently, Visual GPT appears promising for image analysis. It can perform edge and line detection, scene classification, and image segmentation in an intuitive way for unskilled users to make achieving segmentation more easily. However, due to a lack of satellite remote sensing training data, Visual ChatGPT currently under-performs, achieving only 38.1% accuracy for classifying images.[70] This emphasizes an important fact discussed in this paper: using the right tool for the job.

The field of AI will continue to advance, and catch the public and researchers alike off guard with innovations. The temptation to immediately adopt the latest tool can be strong, but a well-planned strategy that takes into account the specific use case is crucial for mission success. AI models come with their strengths and limitations as outlined in this paper, and being able to understand which fits best for any given scenario is the key to incorporating AI in the future of space technologies.

The survey and rubric provided in this paper create a guide to allowing for AI models to be used safely and effectively in spacecraft research. The pairing of AI to a problem must be done thoughtfully and completely. Supervised learning problems need to have sufficient, complete training data. Unsupervised learning problems need clear limitations for unsafe areas so users can predict its actions. AI should not be used without careful consideration, analysis, and a full, technical understanding of the system and the problem it solves.

Acknowledgments

The authors would like to express their gratitude to the NASA Goddard Space Flight Center for their support in providing resources and expertise that helped in the development of this survey and rubric paper. They would also like to thank the MIT Energy Initiative for providing funding through a Seed Grant that helped fund part of this work. They also thank Alex Meredith, Shreeyam Kacker, Madi Anderson, James Staley and Caleb Adams for providing insights and feedback.

References

[1] S. Chien, R. Doyle, A.G. Davies, A. Jonsson, and R. Lorenz. The Future of AI in Space. *IEEE Intelligent Systems*, 21(4):64–69, 2006.

[2] Daniela Girimonte and Dario Izzo. Artificial intelligence for space applications. *Intelligent Computing Everywhere*, pages 235–253, 2007.

[3] C Frost. Challenges and Opportunities for Autonomous Systems in Space. In *Frontiers of Engineering: Reports on Leading-Edge Engineering from the 2010 Symposium*, 2010.

[4] Johns Hopkins University Applied Physics Laboratory. APL AI Technology Roadmap. Technical report, Johns Hopkins University Applied Physics Laboratory, 2020.

[5] M.D. Johnston and G. Miller. Artificial intelligence scheduling for NASA's Hubble Space Telescope. In *[1990] Proceedings. The Fifth Annual AI Systems in Government Conference*, pages 33–39, 1990.

[6] S. P. Synnott and S. D. Desai. The DS-1 Autonomous Navigation System: Autonomous Control of Low Thrust Propulsion System. 1997.

[7] Shyam Bhaskaran, J. Riedel, S. Synnott, and T. Wang. The Deep Space 1 autonomous navigation system - a post-flight analysis. August 2000.

[8] Daniel G. Kubitschek, Nickolaos Mastrodemos, Robert A. Werner, Brian M. Kennedy, Stephen P. Synnott, George W. Null, Shyam Bhaskaran, Joseph E. Riedel, and Andrew T. Vaughan. Deep Impact Autonomous Navigation : the trials of targeting the unknown. Breckenridge, CO, February 2006. NTRS Author Affiliations: NTRS Document ID: 20060043726 NTRS Research Center: Jet Propulsion Laboratory (JPL).

[9] D. Bernard, G. Dorais, E. Gamble, B. Kanefsky, J. Kurien, G. Man, W. Millar, N. Muscettola, P. Nayak, K. Rajan, N. Rouquette, B. Smith, W. Taylor, and Y. W. Tung. Spacecraft Autonomy Flight Experience: The DS1 Remote Agent Experiment. 1999.

[10] G. Rabideau, D. Tran, S. Chien, B. Cichy, R. Sherwood, D. Mandl, S. Frye, S. Shulman, J. Szwaczkowski, D. Boyer, and J. Van Gaasbeck. Mission Operations of Earth Observing-1 with Onboard Autonomy. In *2nd IEEE International Conference on Space Mission Challenges for Information Technology (SMC-IT'06)*, pages 7 pp.–373, 2006.

[11] Steve Chien, Rob Sherwood, Daniel Tran, Benjamin Cichy, Gregg Rabideau, Rebecca Castaño, Ashley Davies, Dan Mandl, Stuart Frye, Bruce Trout, Jeff D'Agostino, Seth Shulman, Darrell Boyer, Sandra Hayden, Adam Sweet, and Scott Christa. Lessons learned from autonomous sciencecraft experiment. In *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS '05, page 11–18, New York, NY, USA, 2005. Association for Computing Machinery.

[12] Patrik Haslum, Nir Lipovetzky, Daniele Magazzeni, and Christian Muise. *An Introduction to the Planning Domain Definition Language*, pages 1–169. Number 2 in Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan and Claypool Publishers, 2 edition, January 2019.

[13] Steve Chien, Joshua Doubleday, David R. Thompson, Kiri L. Wagstaff, John Bellardo, Craig Francis, Eric Baumgarten, Austin

Williams, Edmund Yee, Eric Stanton, and Jordi Piug-Suari. Onboard autonomy on the intelligent payload experiment cubesat mission. *Journal of Aerospace Information Systems*, 14(6):307–315, 2017.

[14] Fernando Figueroa, Mark Walker, and Lauren W. Underwood. NASA Platform for Autonomous Systems (NPAS). In *AIAA Scitech 2019 Forum*, Jan 2019.

[15] Wayne Chi, Jagriti Agrawal, Steve Chien, Elyse Fosse, and Usha Guduri. Optimizing parameters for uncertain execution and rescheduling robustness. *Proceedings of the International Conference on Automated Planning and Scheduling*, 29(1):501–509, May 2021.

[16] Xinwei Wang, Chao Han, Pengbin Yang, and Xiucong Sun. Onboard satellite visibility prediction using metamodeling based framework. *Aerospace Science and Technology*, 94:105377, 2019.

[17] Xianlin Ren and Yi Chen. How can artificial intelligence help with space missions - a case study: Computational intelligence-assisted design of space tether for payload orbital transfer under uncertainties. *IEEE Access*, 7:161449–161458, 2019.

[18] Emmanuel Hebrard, Christian Artigues, Pierre Lopez, Arnaud Lusson, Steve Chien, Adrien Maillard, and Gregg Rabideau. An efficient approach to data transfer scheduling for long range space exploration. In Lud De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 4635–4641. International Joint Conferences on Artificial Intelligence Organization, 7 2022. Main Track.

[19] Dario Izzo, Marcus Märtens, and Binfeng Pan. A survey on artificial intelligence trends in spacecraft guidance dynamics and control. *Astrodynamics*, 3(4):287–299, Dec 2019.

[20] Tamás Vinkó and Dario Izzo. Global optimisation heuristics and test problems for preliminary spacecraft trajectory design. Technical report, 2008.

[21] Matej Petkovic, Luke Lucas, Dragi Kocev, Sašo Džeroski, Redouane Boumghar, and Nikola Simidjievski. Quantifying the effects of gyroless flying of the mars express spacecraft with machine learning. In *2019 IEEE International Conference on Space Mission Challenges for Information Technology (SMC-IT)*, pages 9–16, 2019.

[22] Janhavi H. Borse, Dipti D. Patil, Vinod Kumar, and Sudhir Kumar. Soft landing parameter measurements for candidate navigation trajectories using deep learning and ai-enabled planetary descent. *Mathematical Problems in Engineering*, 2022:2886312, Aug 2022.

[23] Nathan Ré, Timothy Sullivan, Matthew Poplewell, Kirk Roerig, Clayton Michael, Tyler Hanf, and Tyler Presser. Neural Networks for Onboard Maneuver Design. Sept 2022.

[24] A Pellacani, M Graziano, M Fittock, J Gil, and I Carnelli. Hera vision based gnc and autonomy. In *Proceedings of the 8th European Conference for Aeronautics and Space Sciences. Madrid, Spain, 1-4 july 2019*, July 2019.

[25] Charles Dawson, Zengyi Qin, Sicun Gao, and Chuchu Fan. Safe nonlinear control using robust neural lyapunov-barrier functions. In Aleksandra Faust, David Hsu, and Gerhard Neumann, editors, *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pages 1724–1735. PMLR, 08–11 Nov 2022.

[26] Neha Sisodiya, Nitant Dube, and Priyank Thakkar. Next-Generation Artificial Intelligence Techniques for Satellite Data Processing. In D. Jude Hemanth, editor, *Artificial Intelligence Techniques for Satellite Image Analysis*, pages 235–254. Springer International Publishing, Cham, 2020.

[27] Qiang Liu, Jun Yang, Chaojian Zhuang, Ahmed Barnawi, and Bander A Alzahrani. Artificial Intelligence Based Mobile Tracking and Antenna Pointing in Satellite-Terrestrial Network. *IEEE Access*, 7:177497–177503, 2019.

[28] Wenjing Xiao, Rui Wang, Jeungeun Song, Di Wu, Long Hu, and Min Chen. Ai-based satellite ground communication system with intelligent antenna pointing. In *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, pages 1–6, 2020.

[29] Ronald Glumb, Glenn Davis, and Christopher Lietzke. The tanso-fts-2 instrument for the gosat-2 greenhouse gas monitoring mission. In *2014 IEEE Geoscience and Remote Sensing Symposium*, pages 1238–1240, 2014.

[30] Georges Labrèche, David Evans, Dominik Marszk, Tom Mladenov, Vasundhara Shiradhonkar, Tanguy Soto, and Vladimir Zelenevskiy. OPS-SAT Spacecraft Autonomy with TensorFlow Lite, Unsupervised Learning, and Online Machine Learning. pages 1–17, 03 2022.

[31] Georges Labrèche, David Evans, Dominik Marszk, Tom Mladenov, Vasundhara Shiradhonkar, and Vladimir Zelenevskiy. *Artificial Intelligence for Autonomous Planning and Scheduling of Image Acquisition with the SmartCam App On-Board the OPS-SAT Spacecraft*. January 2022.

[32] Georges Labrèche and Cesar Guzman Alvarez. Saasyml: Software as a Service for Machine Learning On-board the OPS-SAT Spacecraft. In *2023 IEEE Aerospace Conference*, pages 1–9. IEEE, 2023.

[33] European Space Agency. Publications overview.

[34] Nassim Abderrahmane, Benoît Miramond, Erwann Kervennic, and Adrien Girard. SPLEAT: SPiking Low-power Event-based ArchiTecture for in-orbit processing of satellite imagery. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–10, 2022.

[35] Shreeyam Kacker, Alex Meredith, Kerri Cahoy, and Georges Labreche. Machine learning image processing algorithms onboard OPS-SAT. August 2022.

[36] Tom Mladenov, Georges Labrèche, Tom Syndercombe, and David Evans. Augmenting Digital Signal Processing with Machine Learning techniques using the Software Defined Radio on the OPS-SAT Space Lab. September 2022.

[37] Erwann Kervennic, Thomas Louis, Michael Benguigui, Frédéric Férésin, Yves Bobichon, and Adrien Girard. Deployment of a cloud segmentation neural network on embedded hardware targets and benchmark of the different deployment toolchains. In *8th International Workshop on On-Board Payload Data Compression (OBPDC 2022)*, pages 1–8, Athenes, Greece, September 2022.

[38] Jakub Nalepa, Jacek Andrzejewski, Bogdan Ruszczak, Krzysztof Kotowski, Alicja Musiał, David Evans, Vladimir Zelenevskiy, Sam Bammens, and Rodrigo Laurinovičs. Toward On-Board Detection of Anomalous Events From OPS-SAT Telemetry Using Deep Learning. In *8th International Workshop on On-Board Payload Data Compression (OBPDC 2022)*, Athenes, Greece, September 2022.

[39] Andrew J. Macdonald, Apeksha Budhkar, Becca Bonham-Carter, Evan Smal, Matt Cross, Kaizad V. Raimalwala, Melissa Battler, and Michele Faragalli. *Enabling Autonomy with a Deep Learning Framework for Planetary Exploration.*

[40] European Space Agency (ESA). The Discovery Campaign on OPS-SAT experiments.

[41] Violet Felt, Shreeyam Kacker, Joe Kusters, John Pendergrast, and Kerri Cahoy. Fast ocean front detection using deep learning edge detection models. *IEEE Transactions on Geoscience and Remote Sensing*, pages 1–1, 2023.

[42] Liya Yuni Astutik and Rahadian Yusuf. Comparing different supervised machine learning accuracy for weather analysis on cubesat simulation. In *2022 International Symposium on Electronics and Smart Devices (ISESD)*, pages 1–5, 2022.

[43] Chandrasekar Radhakrishnan, V. Chandrasekar, Steven C. Reising, and Wesley Berg. Rainfall estimation from tempest-d cubesat observations: A machine-learning approach. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:3626–3636, 2022.

[44] Sedat Yayla and Emrah Harmanci. Estimation of target station data using satellite data and deep learning algorithms. *International Journal of Energy Research*, 45:961–974, 2021.

[45] Seema Mahajan and Bhavin Fataniya. Cloud detection methodologies: Variants and development—a review. *Complex & Intelligent Systems*, 6:251–261, 2020.

[46] Shreeyam Kacker, Alex Meredith, Joe Kusters, Hannah Tomio, Violet Felt, and Kerri Cahoy. On-orbit rule-based and deep learning image segmentation strategies. In *AIAA SCITECH 2022 Forum*, page 0646, 2022.

[47] Frédéric Férésin, Erwann Kervennic, Yves Bobichon, Edgar Lemaire, Nassim Abderrahmane, Gaétan Bahl, Ingrid Grenet, Matthieu Moretti, and Michael Benguigui. In space image processing using AI embedded on system on module: example of OPS-SAT cloud segmentation. Zenodo, jun 2021.

[48] Alan D. George and Christopher M. Wilson. Onboard processing with hybrid and reconfigurable computing on small satellites. *Proceedings of the IEEE*, 106(3):458–470, 2018.

[49] Bradley Denby and Brandon Lucia. Orbital edge computing: Nanosatellite constellations as a new class of computer system. ASPLOS '20, page 939–954, New York, NY, USA, 2020. Association for Computing Machinery.

[50] Caleb Adams, Allen Spain, Jackson Parker, Matthew Hevert, James Roach, and David Cotten. Towards an integrated gpu accelerated soc as a flight computer for small satellites. In *2019 IEEE Aerospace Conference*, pages 1–7, 2019.

[51] Fredrik C. Bruhn, Nandinbaatar Tsog, Fabian Kunkel, Oskar Flordal, and Ian Troxel. Enabling radiation tolerant heterogeneous gpu-based onboard data processing in space. *CEAS Space Journal*, 12(4):551–564, Dec 2020.

[52] Danijela Marković, Alice Mizrahi, Damien Querlioz, and Julie Grollier. Physics for neuromorphic computing. *Nature Reviews Physics*, 2(9):499–510, Sep 2020.

[53] Mike Davies, Andreas Wild, Garrick Orchard, Yulia Sandamirskaya, Gabriel A. Fonseca Guerra, Prasad Joshi, Philipp Plank, and Sumedh R. Risbud. Advancing neuromorphic computing with loihi: A survey of results and outlook. *Proceedings of the IEEE*, 109(5):911–934, 2021.

[54] Marcus Murbach, Alejandro Salas, Michael Lowry, Eric Barszcz, Janette Briones, Peter Schemmel, Michael Mercury, and Tarek Taha. TechEdSat-13: The First Flight of a Neuromorphic Processor, April 2022.

[55] Giacomo Curzi, Dario Modenini, and Paolo Tortora. Large constellations of small satellites: A survey of near future challenges and missions. *Aerospace*, 7(9), 2020.

[56] Thomas Schetter, Mark Campbell, and Derek Surka. Multiple agent-based autonomy for satellite constellations. *Artificial Intelligence*, 145(1-2):147–180, 2003.

[57] Andrew Kitrell Kennedy. *Planning and scheduling for earth-observing small satellite constellations*. Thesis, Massachusetts Institute of Technology, 2018. Accepted: 2019-02-14T15:49:32Z.

[58] Mary Dahl, Juliana Chew, and Kerri Cahoy. Optimization of smallsat constellations and low cost hardware to utilize onboard planning. In *ASCEND 2021*, page 4172. 2021.

[59] Juan Jose Garau Luis, Nils Pachler, Markus Guerster, Inigo del Portillo, Edward Crawley, and Bruce Cameron. Artificial intelligence algorithms for power allocation in high throughput satellites: A comparison. In *2020 IEEE Aerospace Conference*, pages 1–15, 2020.

[60] Sudharman K. Jayaweera and Abriel A. Holland. Deep learning cognitive cooperative data scheduling protocol for small spacecraft swarms. In *2021 IEEE Cognitive Communications for Aerospace Applications Workshop (CCAAW)*, pages 1–5, 2021.

[61] Mengjie Liu, Jian Li, and Hancheng Lu. Routing in small satellite networks: A gnn-based learning approach. *arXiv preprint arXiv:2108.08523*, 2021.

[62] Ali Zolghadri. Advanced model-based fdir techniques for aerospace systems: Today challenges and opportunities. *Progress in Aerospace Sciences*, 53:18–29, 2012.

[63] Daniele Codetta-Raiteri and Luigi Portinale. Dynamic bayesian networks for fault detection, identification, and recovery in autonomous spacecraft. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(1):13–24, 2015.

[64] Aboul Ella Hassanien, Dina Ashraf, and Ahmed Sara. Machine learning in telemetry data mining of space mission: Basics, challenging and future directions. *The Artificial Intelligence Review*, 53(5):3201–3230, 2020.

[65] S. Ramachandran, M. Rosengarten, and C. Belardi. Semi-supervised machine learning for spacecraft anomaly detection and diagnosis. In *2020 IEEE Aerospace Conference*, pages 1–10. IEEE, 2020.

[66] Evana Gizzi, Hayley Owens, Nicholas Pellegrino, Christopher Trombley, James Marshall, and Jivko Sinapov. Autonomous System-Level Fault Diagnosis in Satellites using Housekeeping Telemetry. August 2022.

[67] Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Soderstrom. Detecting spacecraft anomalies using lstms and nonparametric dynamic

thresholding. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining*, KDD '18, page 387–395, New York, NY, USA, 2018. Association for Computing Machinery.

[68] Christopher Tschan, Adnan Yucel, and Nhan Nguyen. *Roadmap For Intelligent Systmes in Aerospace*. American Institute of Aeronautics and Astronautics (AIAA) Intelligent Systems Technical Committee (ISTC), June 2016.

[69] EASA. *EASA Artificial Intelligence Roadmap 2.0*. European Union Aviation Safety Agency, May 2023.

[70] Lucas Prado Osco, Eduardo Lopes de Lemos, Wesley Nunes Gonçalves, Ana Paula Marques Ramos, and José Marcato Junior. The potential of visual chatgpt for remote sensing. *arXiv preprint arXiv:2304.13009*, 2023.