18th International Conference on Space Operations, Montreal, Canada, 26 - 30 May 2025.

Digital Twins and AI Agents for Space Missions

Zhenping Li[a]*

[a] *ASRC Federal, United States, zli@asrcfederal.com*

* *Corresponding Author*

Abstract

The satellite digital twin (SDT) provides model-based data monitoring, engineering analysis, conditional maintenance, and high-fidelity simulation in operations and maintenance. It exceeds current machine learning applications for monitoring satellite health and safety by broadening the scope of satellite telemetry, commands, and orbital event data. The timed finite state machine (TSFM) for satellite operations involving operational events leverages this expanded data scope to form a state equation associating satellite states with operational events in telemetry data and event triggers, such as satellite commands and orbital events. The state equation establishes a framework for more proactive and dynamic monitoring and model-based high-fidelity simulations. The model recalibration in an SDT requires a new architecture to learn state profiles from telemetry data and link these profiles with event triggers to establish the state equation. The model recalibration becomes multi-state data training, which is more effective and accurate in anomaly detection than the existing ML applications. The generative artificial intelligence (AI) agent is proposed as the decision support component in the SDT feedback loop, which leverages the retrieval augmented generation (RAG) to integrate the large language models (LLM) with the satellite knowledge base. The AI agent in SDT offers automated engineer analysis for anomaly resolutions, invokes command procedures for corrective actions for certain satellite states, and serves as a training platform for engineers and operators.

Keywords: Digital Twin, AI Agent, Satellite Operations, Machine Learning, Finite State Machine,

1. Introduction

A Satellite DT (SDT) [1] serves as a virtual representation of space and ground assets within a space mission. The existing infrastructure for space missions facilitates connectivity between an SDT and its physical satellite through ground systems, as all satellites transmit telemetry data regarding their health, safety, and operational status to these systems for monitoring and engineering analysis. An SDT, along with its space and ground assets, establishes a feedback loop where it receives telemetry, mission planning, and log messages for model recalibration, data monitoring, and simulations. Model-based diagnostics and analytics generate corrective actions and recommendations for its ground and space assets to optimize satellite operations. This signifies a paradigm shift from static and statistical monitoring and analysis to model-based dynamic monitoring, high-fidelity simulations, and automated engineering analysis, resulting in reduced risk, optimized operations, and enhanced mission resiliency. Digital twins represent an engineering framework that provides a platform for utilizing AI/ML in dynamic systems across various domains to support decision-making and optimize operations. They have been developed in numerous fields, including healthcare [2,3], agriculture [4], aerospace engineering[5], urban planning and development[6], Earth science[7,8], and Industry 4.0[9,10].

The SDT was proposed during the early stages of digital twin developments [11]. However, little progress was made until recently, with the introduction of a machine learning (ML) application for monitoring satellite health and safety [12,13,14], which developed a framework for training data from satellite telemetry datasets. The initial study[15] of SDTs addresses key challenges in SDT development, including scalability and extensibility in the architecture and efficiency and accuracy requirements for real-time and near real-time model recalibration. One challenge in developing an SDT with an expanded data scope is that a satellite operates as an open and dynamic system, where external commands and events can alter its operational state, rendering the data training approaches in existing ML applications ineffective. Satellite operations involve events like orbital maneuvers, which can last minutes to hours and are initiated by satellite commands or specific orbital occurrences. The data patterns of telemetry during these operational events typically differ from those in standard operations. The purpose of model recalibration in an SDT is to account for the open system dynamics, including operational events triggered by external commands. The TFSM provides a framework for model recalibration in SDTs for open systems, establishing a state equation for satellite operations that integrates telemetry, commands, and orbital event data. This state equation enables model-based satellite simulations and facilitates more proactive anomaly detection. The data models for satellite telemetry are state- and time-dependent, necessitating a new architecture for data training to learn the state equation in satellite operations and to support the training of state-dependent telemetry datasets.

An agent is an artificial entity that perceives its surroundings using sensors, makes decisions, and takes appropriate actions based on predefined mission objectives. It is a crucial component in the SDT feedback loop, receiving outputs from data monitoring or model recalibration regarding satellite operation status and taking appropriate actions to optimize operations. The agent approach in satellite operations is not new; the rule-based agent [16] was developed within the GMSEC framework to facilitate lights-out operations, which has been widely adopted in the industry. It retrieves satellite states from event messages that adhere to the GMSEC[17] standard and command procedures are initiated if these states meet predefined criteria, such as the fail-over procedure in the event of a system component failure. The rule-based agent is simplistic, as it cannot address complex situations requiring understanding satellite states' context and possessing analytical and reasoning capabilities. The emergence of LLMs enables the development of generative AI agents [18] based on LLMs, which provide in-context learning, reasoning, and analytical abilities for a general agent in satellite operations. These agents can serve as a comprehensive knowledge base for space and ground assets in a space mission, broadening the scope of rule-based agents to invoke command procedures for specific satellite states, automate engineering analysis in space operations, offer solutions to engineers for highly complex satellite anomalies, and act as a training platform to respond to any questions related to space or ground assets in a space mission.

This paper expands on the earlier study[15] to address some challenges in the SDT development. These challenges involve integrating generative AI agents within an SDT framework for decision support, the state equation and its accompanying data training architecture in the TFSM, and the recalibration of multi-state models with examples of multi-state training outputs.

2. The AI Agent in an SDT

An SDT[15] consists of a core component for data model recalibration, monitoring, and simulations; a generative AI agent for taking appropriate actions to optimize satellite operations based on the mission objectives and decision support for evaluation by engineers; and a display component for reports, operational status, and AI agent outputs. The SDT core component implements a hierarchical architecture that provides scalability and extensibility. This architecture enables parallel computing to ensure the efficiency of model recalibrations. The output of these recalibrations is the operational status in the event representation, which serves as the input for the AI agents for decision support.

The AI agent plays a vital role in the feedback loop of SDTs for intelligent decision support. It receives log messages from both space and ground assets, and the operational status from the model calibration and monitoring component provides potential solutions for satellite engineers. The agent sends directive requests to the T&CS system to execute operational procedures for immediate action when deemed safe and necessary in case of system failures or anomalies and offers satellite engineers solutions to resolve satellite anomalies. The AI agent integrates an LLM with a local knowledge base for satellite operations using retrieval-augmented generation (RAG)[19]. Data training and monitoring establish satellite states as inputs for the AI agent, combining these states with LLMs' analytical capabilities and the local knowledge base to determine appropriate actions or propose solutions to satellite engineers.

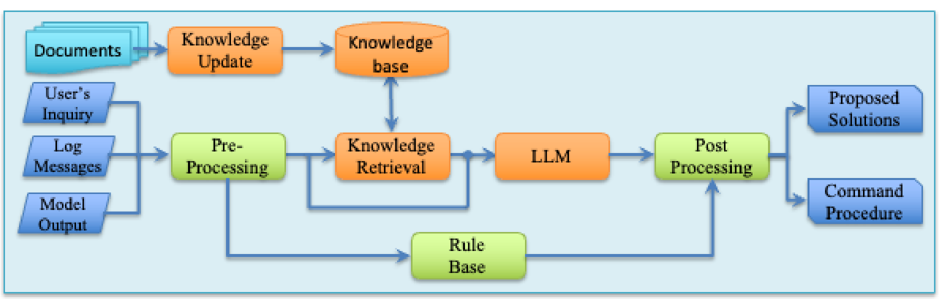

Figure 1 AI Agent Architecture

Figure 1 illustrates the architecture for AI agents involved in satellite operations, generally adhering to the RAG approach[20]. LLMs often have limited or no knowledge of space and ground assets due to the proprietary or classified nature of much relevant domain knowledge regarding satellite operations. This limitation frequently leads to hallucinations in LLM outputs. The RAG process assesses a query to determine if it relates to topics defined in the knowledge base. If it does, the process retrieves information relevant to the user's question from the knowledge base. The appropriate context from the knowledge base is then provided to the LLM and the original query, which helps reduce hallucinations in the resulting LLM response. The agent also includes a rule base for the mature and well-defined command procedures in response to certain satellite states detected from log messages or model outputs. As it runs in real-time environments, its operational efficiency is essential. The direct inquiry into the rule base without inference to LLMs makes the agent operations much more efficient. The rule base within the agent is the same as that specified in Ref. 19, demonstrating its high efficiency in real-time operational environments.

The input documents for the satellite knowledge base within the AI Agent consist of hardware and software manuals, user manuals for both space and ground assets, metadata for command procedures, a satellite command and telemetry database, and any documents related to satellite operations. These documents empower AI agents to troubleshoot anomalies and plan corrective actions by invoking the appropriate command procedures or recommending solutions to satellite engineers. The knowledge base must be adaptive to the emerging states or issues in satellite operation with manual or automatic updates, ensuring that the AI Agent stays current. Command procedures could also be incorporated into the rule base for emerging states in space and ground assets.

The real-time inputs to the AI agent include log messages from the mission enterprise and outputs from the data recalibration and monitoring component in an SDT. These log messages provide the operational status of both space and ground assets. They may also include keep-alive messages from components in the ground systems to support failover operations in case of a component or subsystem failure, which is crucial for lights-out operations. The model recalibration and monitoring component outputs in an SDT represent the satellite's operational state. In anomalous conditions during satellite operations, which often involve unexpected changes in data patterns across various datasets in multiple subsystems, the AI agent analyzes the causes of the anomalies and provides solutions for engineers. The agent also serves as a chatbot to provide training and a help platform for engineers, who can look for answers on hardware, software, and operation procedures in satellite operations.

The agent's outputs include a module for invoking command procedures, a chatbot-like display for the LLM's responses to user inquiries, and potential solutions to anomalies for engineers and managers to review. Anomalies in satellite operations rarely occur, and each has unique characteristics. The solutions generated by the agent must be reviewed by engineers and managers, allowing humans to be in the middle of the feedback loop in SDTs. The agent in SDTs doesn't directly invoke command procedures; instead, it sends requests to the ground system's telemetry and command system (TCS) to initiate a command procedure. Thus, an enterprise ground system architecture with the message standard must enable the plug-and-play SDT components, send directive requests to the TCS component, and receive real-time log messages.

4. The State Equation for Satellite Operations

A satellite is an open system, and its states are not static; they are influenced by operational events, mission-specific payload activities, and space weather conditions. Examples of operational events include orbital maneuvers or momentum dumps initiated by satellite commands. Another example is the eclipse state, triggered by an orbital event in which the satellite moves behind the Earth, resulting in changes to the power and thermal subsystems for geosynchronous satellites. These operational events can last from a few minutes to several hours before normal operations resume, and satellite telemetry datasets during these events exhibit different patterns compared to those during normal operations.

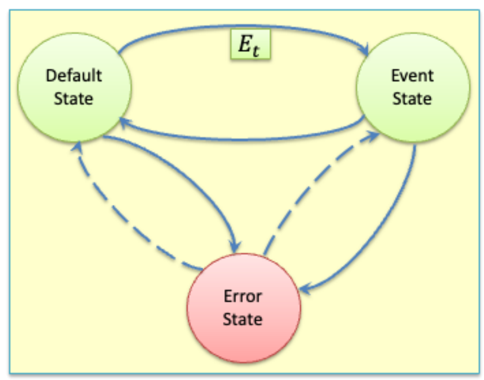

Figure 2 The Timed Finite State Machine for satellite operations.

An open digital twin interacts with its environment and changes its state with external commands or events in its environment. An SDT is an open digital twin and can be regarded as a finite state machine, as shown in Figure 2. A satellite in normal operations without operational events or anomalies is defined as a default state. Operational events triggered by commands or external events are defined as the event states independent from the default state in TFSM. Anomalies are error states independent of both default and event states. The triggers for transitions between error states and default or event states can be regarded as the root causes of anomalies, which are generally unknown and require engineering analysis. The update function λ in the TFSM is the state equation for satellite operations that defines the transition from a default state to an event state or an anomalous state

$$\lambda: C \ \rightarrow \{q_l^e, f_k^e(\delta t)\}. \tag{1}$$

It shows that a state for an operational event in a period $\{t_i^e, t_f^e\}$ is triggered by a trigger C at the time t_i^e from a default state. The variable $\{q_l^e, f_k^e(\delta t)\}$ represents a state file comprised of discrete datasets q_l^e and continuous datasets $f_k^e(\delta t)$ where $\delta t = t - t_i^e$. The Trigger C in Eq. 2 comes from satellite commands or orbital events. The state equation in Eq. 2 associates satellite commands and orbital events with telemetry data. It is equivalent to a rule model: a satellite is in the state $\{q_l^e, f_k^e(\delta t_e)\}$ if trigger C is true, it creates situational awareness in satellite operations since one can anticipate changes in the satellite telemetry with the incoming triggers.

The state equation lays the groundwork for model-based data monitoring and simulations. A known satellite state is always linked to a satellite command or an orbital event. A state lacking a known trigger may indicate a potential anomalous state. Model recalibrations can establish the association of satellite commands or orbital events with satellite states. Thus, one can anticipate the satellite state with the incoming trigger, making model-based data monitoring more proactive and accurate. On the other hand, a satellite command from a ground system in simulations triggers a transition from a default state to an event state by generating satellite telemetry and sending it back to ground systems. The same set of data models in the data monitoring is used in satellite simulations: the data model monitoring process compares the data value to the model predictions, while the model-based satellite simulations generate telemetry data from model predictions.

5. The Model Recalibration in an SDT

The state equation in the TFSM framework requires a new architecture for model recalibrations. Fig. 3 shows a training architecture for learning the state equation in Eq. 1 that integrates event triggers with telemetry datasets. The model recalibration is performed in sessions with the iterative training approach[15]. The model input in Fig. 3 is the data model created in the previous session to carry out initial outlier detection and to serve as the input for the data training of the current session. The model recalibration generates the state profile, $\{q_i^e, f_k^e(\delta t_e), \sigma_j\}$, and detects the outliers. To establish the state equation, the post-training process correlates the state files with event triggers, such as satellite commands or orbital event data. The post-training process also implements the clustering algorithm for anomaly detection.

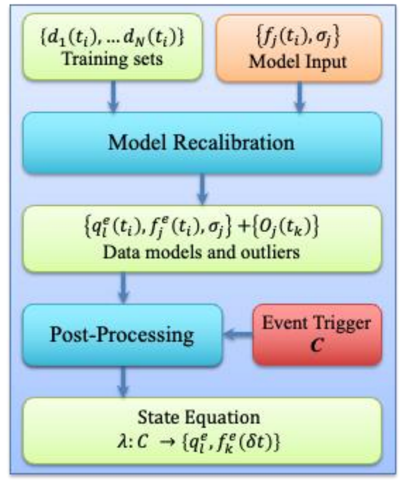

Figure 3 The architecture for learning the state equation in satellite operations

The model recalibration involves rule, data-driven, and hybrid models. Ref. [15] discusses examples of hybrid models and their data training algorithms. This session will focus on discrete and data training models with multiple states during training periods. The discrete datasets can be written as a time-dependent function

$$q_j^d(t) = q_j^d(t_0) + \sum_k \delta_j^k(t_k). \tag{2}$$

Data training for discrete datasets involves a straightforward search process that detects the change value $\delta_j^k(t_k)$ data at a specified time t_k. Rule models for discrete datasets are obtained by correlating the change value $\delta_j^k(t_k)$ in Eq. 1 with an event trigger *C* at the specified time t_k.

A data-driven model for a dataset $\{d_j(t)\}$ is represented by a time-dependent trend consisting of a time-dependent function $f_j(t - t_0)$ and a standard deviation σ_j expressed as

$$\sigma_j = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(d_j(t_i) - f_j(t_i - t_0)\right)^2} \tag{3}$$

representing its noise level, where t_0 is the reference time. Since the period for model recalibrations generally covers 2 to 4 orbital periods, multiple states in a training period are possible. Hence, the recalibration of data-driven models is a data training of multivariable functions. A state flag can characterize an event state in satellite operations:

$$q_j^e(t) = \begin{cases} 1, & t_i^e \le t < t_f^e \\ 0, & t < t_i^e \, or \, t \ge t_f^e \end{cases}, \tag{4}$$

in the period $\{t_i^e, t_f^e\}$. The multi-state function $f_k(\delta t, q_k^e)$ in the data training period can be approximated as

$$f_k(t, q_k^e) = (1 - q_k^e) f_k^d(\delta t_0) + q_k^e f_k^e(\delta t_e) \tag{5}$$

where $\delta t_0 = t - t_0$ and t_0 is a reference time for the time-dependent function $f_k^d(\delta t_0)$ in default states. Thus, the model recalibration for the multi-state function $f_k(\delta t, q_k^e)$ involves data training for the function $f_k^d(\delta t_0)$ of the default state and the function $f_k^e(\delta t_e)$ for event states, and each state during the training period has a well-defined period $\{t_i^e, t_f^e\}$. The models for default state $f_k^d(\delta t_0)$ and event state $f_k^e(\delta t_e)$ are generally different.

The data trainings are performed separately for different states. Figure 4 shows an example of the multi-state training outputs. The mnemonic is the charge state for the battery system in a geosynchronous satellite. The yellow-shaded period covers the Sun-Earth eclipse state, in which the satellite moves behind the Earth so that the battery system cannot be charged through sunlight, reducing the charge state. The eclipse state happens twice yearly and lasts from minutes to more than an hour, changing the data patterns in mnemonics of power and thermal systems. The data pattern changes slowly as the period of the eclipse state changes. The data in the green-shaded period are in the maneuver state, showing a dip in data. The maneuver state is more dynamic and could occur at any time, and the period for maneuver states lasts from minutes to about an hour. The mnemonics in the power, thermal, and GNC subsystems are affected by maneuver states. The periods not covered by green or yellow shade belong to the default state, which shows a constant value. The default, eclipse, and maneuver states generally have different data models. The data model for the default state is a simple statistical model, while the data models for the eclipse and maneuver states are the

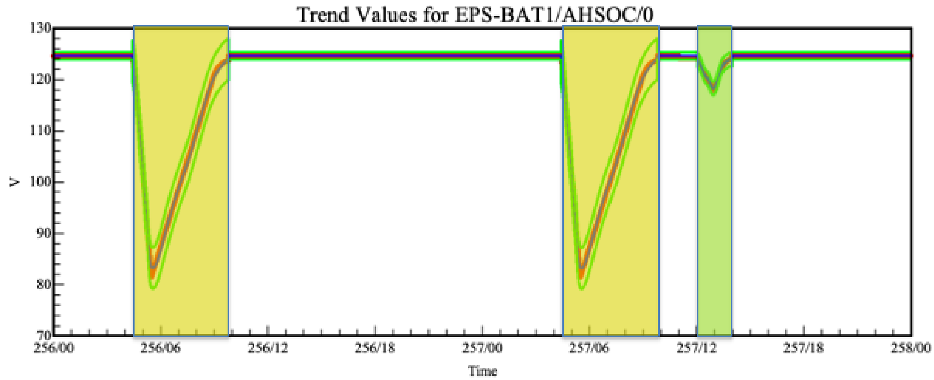

Figure 4 The multi-state training outputs for the charge state in the battery system. The data in the yellow-shaded period are in the eclipse state, and those in the green-shaded period are in the maneuver state. The grep lines are the data bound defined in Eq. 1, the red dots are the data value, and the blue line is the model output.

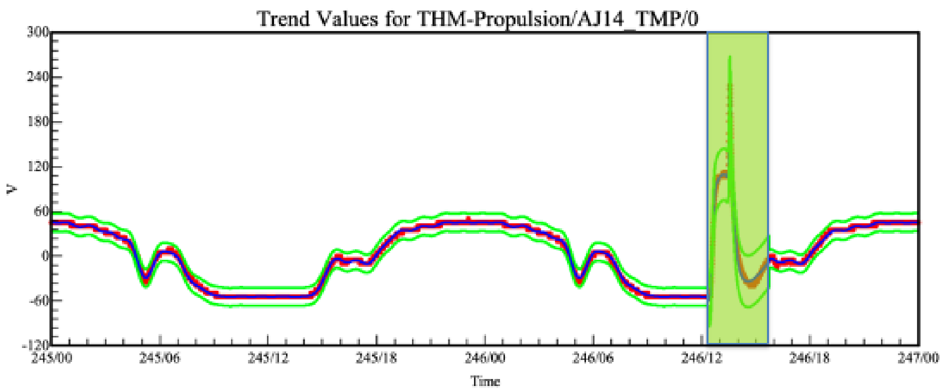

Figure 5 The temperature mnemonic in the propulsion subsystem. The green-shaded period corresponds to the maneuver state in which the thruster is fired to cause the temperature spikes. The green lines represent the data bound of the model output. The red dots are the telemetry data points, and the blue line is the model output.

neural networks. The standard deviations that determine the data bound (green line) are generally different since each state has its data training. Figure 4 shows a larger standard deviation for the eclipse state.

Figure 5 shows an example of the temperature mnemonics in the propulsion subsystem. The green-shaded period covers the maneuver in which a thruster is fired, elevating the temperature. A geosynchronous satellite has multiple thrusters, but only a subset of thrusters is used in a housekeeping maneuver. Thus, the maneuver flag alone cannot determine if a thruster is in the maneuver state, and the combination of the maneuver flag and the thruster enable flag is used to determine if a thruster is active in the maneuver state, leading to the temperature spike in Figure 5.

The data points in the maneuver state are outliers in the existing ML application for monitoring satellite health and safety telemetry, which trains data only for the default state. However, the outliers associated with the maneuver state do not correspond to anomalies. Distinguishing outliers tied to an operational event from anomalies has been a significant challenge for the current ML application.

Figure 6 presents an example of a mnemonic within the GNC subsystem featuring multiple satellite states. Operational events that alter the satellite GNC telemetry include maneuvers and momentum adjustments. Maneuver events position the satellite optimally, while momentum adjustment events reset the reaction-wheel subsystem necessary for controlling satellite attitudes. Figure 6 illustrates four events with shorter durations related to the momentum adjustment events and one with a longer duration related to the maneuver events. The data models for the default, maneuver, and momentum adjustment states are the statistical model, and the standard deviations for maneuver and momentum adjustment are significantly elevated, which is a typical feature in the telemetry data for geosynchronous satellites.

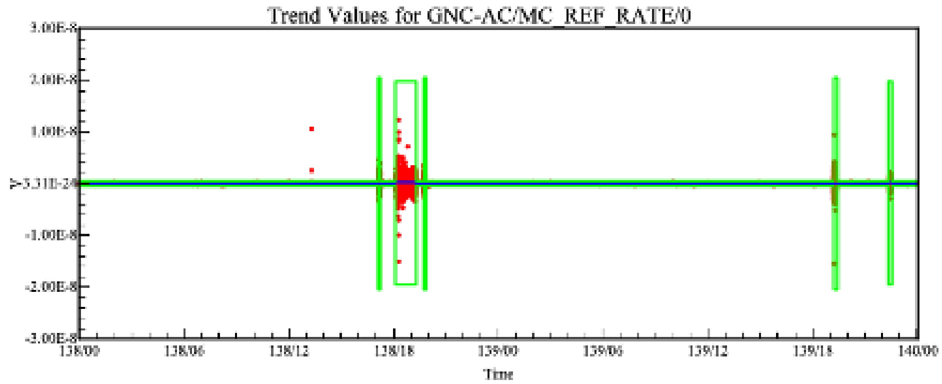

Figure 6 A mnemonic in the GNC subsystem affected by the maneuver and momentum adjustment.

Model recalibration using multiple states represents a significant advancement over the machine learning application that only trains data for default states. This approach eliminates false outliers during operational events in the existing machine learning approach. Multi-state model recalibration enhances the robustness and accuracy of anomaly detection and enables model-based satellite simulations.

6. Summary

This paper presents solutions for critical challenges in developing SDTs using innovative technologies. Satellite operations are open and dynamic systems whose datasets are influenced by external events and commands. The state equation in TFSM links profiles of satellite states with event triggers, enabling the identification of anomalous states with unknown triggers as outliers. This method enhances anomaly detection in an SDT, making it more proactive and accurate while eliminating false positives from operational events. Additionally, the state equation in TFSM promotes a new learning architecture that correlates state profiles with event triggers and requires multi-state training instead of single default state training in the existing ML applications. Multi-state training greatly improves model-based data monitoring by removing outliers caused by operational events in single-state data training, leading to more proactive and accurate anomaly detections.

The generative AI agent plays an essential role in an SDT feedback loop. The data recalibration and monitoring component evaluates the satellite's operational status, which serves as input for the agent. The agent analyzes local satellite knowledge to present solutions or actions to engineers or automatically takes appropriate measures. The emergence of LLMs enhances the agent’s capabilities in natural language processing, reasoning, and planning while integrating LLMs with a comprehensive satellite knowledge base through the RAG framework ensures the reliability and accuracy of the agent’s actions and the proposed solutions.

The SDT’s data recalibration and monitoring component has been developed and deployed in operations. The implementation of the generative AI agent is in progress. Our investigation shows that digital twins offer a natural platform for applying AI/ML technologies to automate and optimize satellite operations.

Acknowledgments

The author wishes to express his gratitude for the support the engineering team at ASRC Federal in deploying the software into operations.

References

[1] Digital Twin: Definition & Value — An AIAA and AIA Position Paper (AIAA, 2021); https://www.aia-aerospace.org/publications/digital-twin-definition-value-an-aiaa-and-aia-position-paper.

[2] Kaushik P. Venkatesh, Gabriel Brito, and Maged N. Kamel Boulos, “Health Digital Twins in Life Science and Health Care Innovation”, Annual Review of Pharmacology and Toxicology, vol 64 (2024) 159. doi:10.1146/annurev-pharmtox-022123-022046.

[3] Laubenbacher, R., Mehrad, B., Shmulevich, I. *et al.* “Digital twins in medicine”. *Nature Computational Science* vol 4 (2024), 184–191. doi:10.1038/s43588-024-00607-6.

[4] Peladarinos N, Piromalis D, Cheimaras V, Tserepas E, Munteanu RA, Papageorgas P. “Enhancing Smart Agriculture by Implementing Digital Twins: A Comprehensive Review”. *Sensors*. vol 23(16), (2024) 7128. doi:10.3390/s23167128

[5] Ferrari, A., Willcox, K. Digital twins in mechanical and aerospace engineering. Nature Computational Science Vol 4, (2024) 178–183. doi:10.1038/s43588-024-00613-8

[6] Syed Abdul Rahman, S. A. F., Abdul Maulud, K. N., Ujang, U., Wan Mohd Jaafar, W. S., Shaharuddin, S., & Ab Rahman, A. A. (2024). The Digital Landscape of Smart Cities and Digital Twins: A Systematic Literature Review of Digital Terrain and 3D City Models in Enhancing Decision-Making. Sage Open, vol 14, (2024) 1. doi:10.1177/21582440231220768

[7] Bauer, P., Stevens, B., Hazeleger, W. “A digital twin of Earth for the green transition.” Nature Climate Change V11, (2021), 80–83. doi:10.1038/s41558-021-00986-y.

[8] J. Le Moigne, “NASA’s Advanced Information Systems Technology (AIST): Combining New Observing Strategies and Analytics Frameworks to Build Earth System Digital Twins (ESDT),” IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium, Kuala Lumpur, Malaysia, (2022), 4724-4727. doi:10.1109/IGARSS46834.2022.9883640

[9] Jiewu Leng, Dewen Wang, Weiming Shen, Xinyu Li, Qiang Liu, Xin Chen, “Digital twins-based smart manufacturing system design in Industry 4.0: A review”, Journal of Manufacturing Systems, vol 60, (2021), 119-137. doi:10.1016/j.jmsy.2021.05.011.

[10] Tao, F., Zhang, H. & Zhang, C. Advancements and challenges of digital twins in industry. Nature Computational Science Vol. 4, (2024), 169–177. doi:10.1038/s43588-024-00603-w

[11] Glaessgen, Edward, and David Stargel. "The digital twin paradigm for future NASA and US Air Force vehicles." In 53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics, and Materials Conference 20th AIAA/ASME/AHS Adaptive Structures Conference 14th AIAA (2012), 1818.

[12] Zhenping Li, "Machine Learning in Spacecraft Ground Systems," 2017 6th International Conference on Space Mission Challenges for Information Technology (SMC-IT), Madrid, Spain, (2017),76-81, doi:10.1109/SMC-IT.2017.20.

[13] Zhenping Li, “A Machine Learning Solution for Satellite Health and Safety Monitoring,” Journal of Satellite Operations & Communicator, Jan. 2022. https://www.opsjournal.org/DocumentLibrary/Uploads/SatOpsML-paper-zli%20(002)%20AG_final%20.pdf.

[14] Zhenping Li “Revolutionize Satellite Health and Safety Monitoring with Advanced Intelligent Monitoring System” ASRC Federal White Paper, August 2023, doi:10.13140/RG.2.2.20068.55686.

[15] Zhenping Li, “Satellite Digital Twins”, Journal of Satellite Operations & Communicator, July, 2024, https://opsjournal.org/DocumentLibrary/Uploads/AIAA-paper-DT-ZPLI.pdf

[16] Zhenping Li and C. Savkli, "Autonomic computing for spacecraft ground systems," *2nd IEEE International Conference on Space Mission Challenges for Information Technology (SMC-IT'06)*, Pasadena, CA, 2006, pp. 8 pp.-520, doi: 10.1109/SMC-IT.2006.21.

[17] See https://github.com/nasa/GMSEC_API.

[18] Xi Z H, Chen W X, Guo X, et al. “The rise and potential of large language model based agents: a survey.” Sci China Inf Sci, 2025, 68(2): 121101, https://doi.org/10.1007/s11432-024-4222-0

[19] Lewis, Patrick, et al. "Retrieval-augmented generation for knowledge-intensive nlp tasks." Advances in Neural Information Processing Systems 33 (2020): 9459-9474.