# Hardware-In-The-Loop Tests of Complex Control Software for Rocket Propulsion Systems

Conference Paper · October 2020



4 authors:

Günther Waxenegger-Wilfing
German Aerospace Center (DLR)


Kai Dresia
German Aerospace Center (DLR)


M. Oschwald
German Aerospace Center (DLR)


Klaus Schilling
University of Wuerzburg

IAC–2020–C4.1.15

# Hardware-In-The-Loop Tests of Complex Control Software for Rocket Propulsion Systems

**Günther Waxenegger-Wilfing**
DLR (German Aerospace Center), Germany, guenther.waxenegger@dlr.de

**Kai Dresia**
DLR (German Aerospace Center), Germany, kai.dresia@dlr.de

**Michael Oschwald**
DLR (German Aerospace Center), RWTH Aachen University, Germany, michael.oschwald@dlr.de

**Klaus Schilling**
University Wuerzburg, Germany, schi@informatik.uni-wuerzburg.de

Software of rocket propulsion systems has a very high complexity but must also guarantee an outstanding level of reliability. Therefore, extensive tests are inevitable during the development process to meet the requirements. However, ground tests of the entire propulsion system are extremely expensive and can not address all nominal and off-nominal behavior that could possibly occur during the mission. Hardware-in-the-loop simulations allow realistic testing of embedded systems (e.g., the engine controller or an actuator) by combining a software simulation with emulated electrical sensor and actuator signals. This contribution discusses hardware-in-the-loop methods for anomaly detection and control systems of rocket propulsion systems. We develop a roadmap surveying a step-by-step approach to demonstrate the reliability of advanced control algorithms and sketch how to deploy such algorithms on space-graded embedded systems. In the future, adaptive and reconfigurable control for the propulsion system will become increasingly important. Adaptive control can tune the control actions to uncertain or changing system characteristics. Also, coupling the control algorithms to suitable anomaly detection systems allows the safe reaction to unforeseen events, which further improves the reliability of the propulsion system. The DLR Institute of Space Propulsion operates extensive test facilities for developing propulsion system technologies to operational maturity and ensuring their quality. Specific attitude pointing aspects of the propulsion system software can be simulated by high precision turntables at the company Zentrum für Telematik (ZfT). Within the cooperation between the DLR Institute and ZfT, different aspects of complex control software for propulsion systems are analyzed and tested using the available facilities.
**keywords:** hardware-in-the-loop simulation, rocket engine control, control system development, rocket propulsion test facilities

## 1. Introduction

The demands placed on the software are constantly increasing in the area of space transportation. The reasons for this are that mission profiles are becoming more complex, e.g. landing of first stages for subsequent reuse, and that control aspects are gaining in importance.[4]

Liquid rocket engines like the Space Shuttle main engine use closed-loop control at most near steady operating conditions. The control of the transient phases is traditionally performed in open-loop due to highly nonlinear system dynamics. The situation is unsatisfactory, in particular for reusable engines. The open-loop control system cannot react to external disturbances. It is therefore intended to extend the use of closed-loop control to the transient phases. Only optimal control can guarantee a long life expectancy of the engine without damaging pressure and temperature spikes. Although the importance of a suitable closed-loop control system has been evident for many years, the majority of rocket engines still employ valves which are operated with pneumatic actuators, too inefficient for sophisticated closed-loop control. The development of an all-electric control system[3] started in the late 90s in Europe. The future European Prometheus engine will have such a system.[16] Other countries are also well advanced in the research and development of electrically operated flow control valves.[1] Due to the electrification of the actuators and the grown demands, the interest in optimal engine control increased recently and will further rise in the future.

Since rocket engines operate at the limits of what is technically feasible, they are inherently susceptible to anomalies. The immense costs associated with the loss of the launch vehicle clearly show the importance of a suitable condition monitoring system. Machine condition monitoring systems must provide a proper diagnosis in real-time from existing sensor data to detect abnormal behavior and, for example, to trigger an emergency shutdown. Condition monitoring algorithms must be able to detect all critical faults that may occur during the operation of a rocket engine. Furthermore, it is not enough to detect and categorize a fault. One must also be able to react accordingly to it. Fault-tolerant control systems aim at an optimal control reconfiguration and satisfy the industrial demand for enhanced availability and safety, in contrast to traditional reactions to faults, which bring about sudden shutdowns and loss of availability.

Modern control methods offer significant advantages compared to classical techniques. However, the use in a safety-critical area such as space transportation places great demands on robustness and reliability. For this reason, the testing of appropriate software is of utmost importance.

## 2. Rocket Propulsion Systems Control

Rocket propulsion systems produce thrust by ejecting a gas that has been accelerated to high speed through a propelling nozzle. Efficient engines use the combustion of reactive chemicals, consisting of fuel and oxidizer components, within a combustion chamber to supply the necessary energy. According to Newton's third law, the gas expansion pushes the engine in the opposite direction. The net thrust of a rocket engine depends on the mass flow of the exhaust gas and its effective exhaust velocity. Changing the direction of the thrust can be achieved by gimbaling the whole engine, which is the standard procedure for big liquid rocket engines used on modern launch vehicles. For a given propellant combination, nozzle geometry, and ambient conditions, the magnitude of the thrust is a function of the combustion chamber pressure and mixture ratio, i.e. oxidizer to fuel ratio. Thus, those factors are usually the focus of the main control loops and are controlled via adjustable flow control valves.[12] Linear actuators are used to change the gimbal angle of the engine and therefore the direction of the thrust, compare Fig. 1. Further actuators are the ignition systems. Sequences like the start-up of a rocket engine must be carried out in a closely coordinated manner. Pressure peaks may cause backflows in the feed lines, which can destroy the engine. Other constraints are given by temperature and rotational speed limits. Especially the reuse of engines necessitates the use of condition monitoring methods. This is the only way to monitor changes in system health. Condition monitoring and anomaly detection are also a prerequisite for health management techniques and emergency shutdowns. As already mentioned, it is common to handle the demanding transients in open-loop. In steady-state operation, classic control methods, e.g. of the PI type, are used if at all. Concerning the condition monitoring simple red line methods are state-of-the-art. Although classical control algorithms are well tested and can be made very robust, they have fundamental limitations and may not be sufficient for future reusable engines in terms of performance. To make matters worse, the performance of a reusable engine might degrade over time due to soot depositions, increased leakage mass flows, which are caused by seal aging, or turbine blade erosions. Therefore, it must be possible to easily adapt the control system to guarantee optimal performance. The last years have witnessed an enormous interest in the use of artificial intelligence methods for automatic control, especially machine learning algorithms including neural networks. These methods seem to be

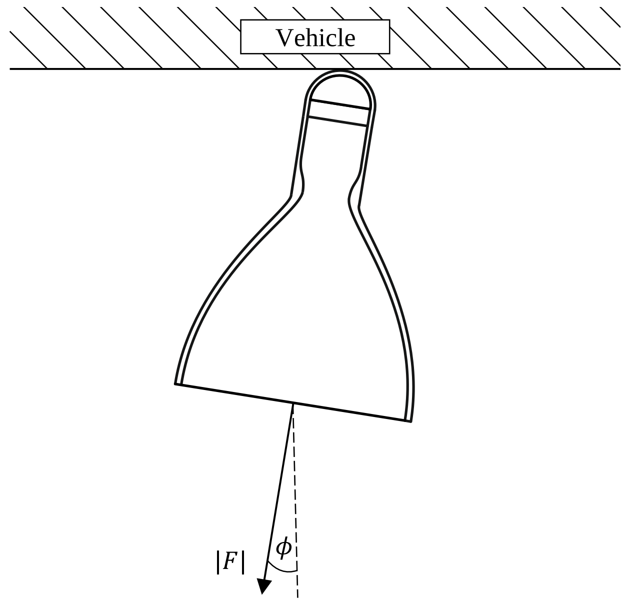


Fig. 1: Gimbaled Engine

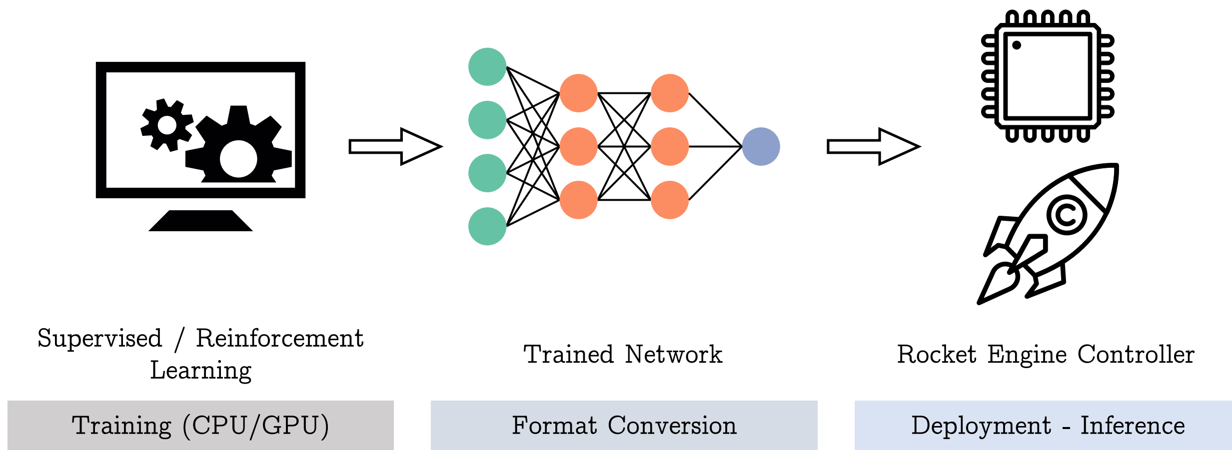


Fig. 2: Development Flow. Figure adapted from Fig. 1 by Nadeski[11]
.

particularly promising for adaptive control, fast optimal control, and condition monitoring, but challenges remain.

A major challenge in rocket engine applications is that the control systems are safety-critical. A rocket engine can destroy or degrade itself and the environment around it if improperly controlled. Reliability and safety have top priority in aerospace applications. Considering operation constraints is fundamentally necessary in such cases. Constraints are not only important during system operation, but also during adaption phases as well. Although the stability of neural network controllers is in general not guaranteed, there has been promising recent work on certifying the stability of certain control policies.[8] The verification and validation of neural networks and adaptive systems have intensively been examined as well.[7] Another challenge is given by the limited computational resources associated with typical embedded systems used in space applications.

## 3. Deployment on Embedded Systems

As precisely stated by Roth et al.,[13] the computing power during the operating phase of machine learning algorithms is typically limited. This is particularly true for space applications, where robust and fail-safe computing hardware is used in the harsh environment of space or during a rocket launch. Furthermore, high-performance computing hardware with high weight is not feasible for space launch vehicles as weight is one of the most critical design parameters limiting the launcher's performance. In recent years, different techniques were developed to reduce the computational demands during neural network training and inference.[13] These techniques can reduce memory usage, and increase inference speed and energy efficiency.

However, these improved methods are only necessary for very deep and complex neural network architectures used for computer vision (e. g., object detection or video tracking), natural language processing, or when training the neural network directly on the embedded system. For most applications, it is enough to deploy only the trained neural network on the embedded system. One would train the neural network using suitable data sets (supervised learning) or simulation environments (reinforcement learning) on a dedicated workstation in Python with commonly used machine frameworks such as TensorFlow or PyTorch. Then, one would convert the neural network to C/C++, and copy the network to the embedded system on the actual space system, compare Fig. 2. Thus, the embedded system just needs to handle the inference, which has much lower computational demands than the training of the network.

Furthermore, the necessary network architectures for engine control and condition monitoring are much smaller than those used for computer vision. For example, a closed-loop controller of the combustion chamber pressure and mixture ratio was realized with only two hidden layers of 400 and 300 neurons.[19] A single inference on this network needs only approximately 250 000 floating-point operations (flops). A space-graded microcontroller used in a variety of spacecraft is the RAD750 with a computing performance of 80 million flops per second.[10] Clearly, this microcontroller can handle the inference in real-time.

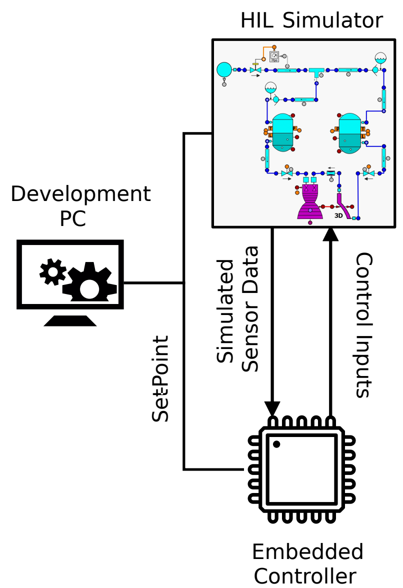


Fig. 3: HIL Configuration

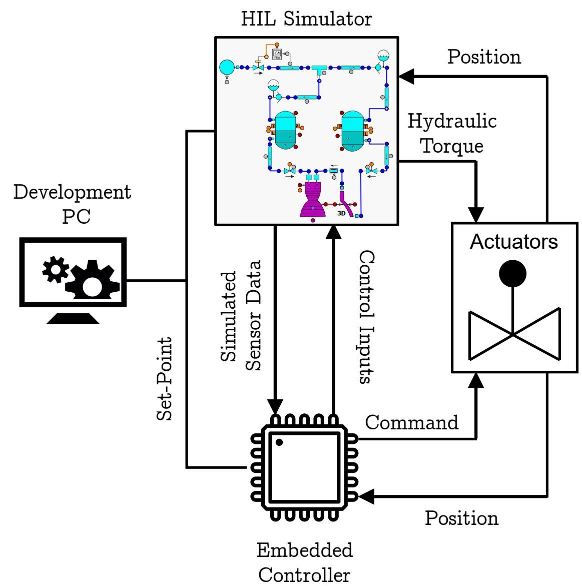


Fig. 4: Extended HIL Configuration

For computer vision or similar applications, which need bigger neural networks and more computing performance, rugged computers based on industrial AI boards could be used after strengthening them for space conditions.

There is a loss in accuracy if a neural network is trained only in a simulated environment. Especially when aerodynamics and fluid dynamics have to be considered. There are reinforcement learning approaches that explicitly consider modeling errors. Domain randomization can produce controllers that generalize well to a wide range of environments.[17] Nevertheless, fine tuning of the control law on the real propulsion system is preferable. Since the test bench measurement command and control systems lacks corresponding restrictions on computing power, such a procedure would be executed using test bench infrastructure. For control law adaption one could also use recorded flight data.

## 4. Hardware-in-the-Loop Testing

Hardware-in-the-loop (HIL) testing[2] refers to the inclusion of physical devices (hardware) into the simulation process. A typical example would be connecting a real physical controller, i.e., the same one that would perform this duty in the actual plant, with a software simulation of the plant. This allows validating a control system without putting the plant (in our case the propulsion system) at risk. It is not only more economical to develop and test while connected to a HIL simulator than the real plant, but also increases the scope of the testing. HIL tests enable testing at or beyond the range of the nominal engine parameters and verification of the system at failure conditions. A suitable mathematical system representation is required, which must be real-time capable.

A suitable simulation environment for rocket propulsion systems is given by EcosimPro, which is a modeling and simulation tool for 0D and 1D multidisciplinary continuous and discrete systems. In EcosimPro, the system description is based on differential-algebraic equations and discrete events. Within a graphical user interface, one can combine different components, which are arranged in several libraries. Of particular interest are the European Space Propulsion System Simulation (ESPSS) libraries, which are commissioned by the European Space Agency (ESA). These EcosimPro libraries are suited for the simulation of liquid rocket engines and test benches. Employees of DLR Lampoldshausen have many years of experience in the use of EcosimPro/ESPSS and used it for example for the design and optimization of the LUMEN engine.[5] Furthermore, advanced machine learning based surrogate models can be integrated into EcosimPro to improve the simulation quality.[6,18]

Using an EcosimPro deck, i.e. is a simulation model designed to run as a standalone black box independent from the main program, and so called S-functions the model can be easily exported to Simulink. Via the Simulink Coder tool, a package is obtained that is ready for use on different platforms, which can act as HIL simulators like the National Instruments PXI platforms. It must be checked that real-time conditions are satisfied in the execution of the model selecting an appropriate sample time.

Fig. 3 shows a HIL set up which includes an electrical emulation of all sensors and actuators. The embedded controller receives simulated sensor data from the HIL simulator and sends control inputs to the HIL system.

As a next step, real actuator hardware can be included. Electrically operated flow control valves are composed of three parts: the hydraulic part, the actuator part and the electronics command part. The actuators, which can be given by one or two electrical motors, can easily be included in the HIL tests, compare Fig. 4. The hydraulic parts have to be replaced by other actuators which provide a realistic torque calculated by the HIL simulator. Obviously, not all actuators can easily be included in this way, e.g. ignition systems are still emulated. A more complex test set up also allows the partial inclusion of representative environmental conditions, e.g., temperature and vibrations.

The HIL simulator can also simulate the occurrence of anomalies and the gradual deterioration of system health. Therefore this system is perfectly suited for testing the anomaly detection and condition monitoring functions of the developed real engine controller.

## 5. DLR Rocket Engine Test Facilities

Although HIL tests can make a significant contribution to the development of rocket engine control systems, they cannot completely replace tests of the control system during the firing of a real engine using a suitable rocket engine test facility. The main reason is that even the most sophisticated simulation models have prediction errors due to not included effects or model miss-specification. This is especially true for complex systems like rocket engines. Thus, real engine tests are needed for system identification and tuning of the controllers. Fig. 6 shows the development process for a new engine controller. In the first step, the implemented control software is tested within a modeling environment to prove the control software design (software-in-the-loop testing). Next, the previously discussed hardware-in-the-loop tests are performed to test the embedded control architecture. Finally, the developed control system and the simulation models are validated by firing tests of a real engine at a suitable rocket engine test facility.

Fig. 5: P8.3 Engine Test Facility

Sensor data from hot firing tests also provide the training data for machine learning based condition monitoring systems. Fortunately there are machine learning algorithms, which only need data for nominal operation. Once trained, neural networks can be used to distinguish nominal and off-nominal behavior. By taken the correlations and interactions between various variables into account, the transition to off-nominal behavior is detected before a classical red line gets triggered.

The test facilities at the DLR Institute of Space Propulsion are a fundamental pre-requisite for developing engine technologies to operational maturity and ensuring their quality. The P4 altitude simulation test bench played a indispenable role during the development of the Vinci upper stage engine, which will be on the upper stage of the Ariane 6 launch vehicle. The P5.2 test facility can even be used to qualify not only engines and individual components, but also complete cryogenic upper stages. Perfectly suitable for researching advanced technologies for future space engines is the P8 test bench, which is operated by CNES, DLR and ArianeGroup. At the interfaces to the test objects in the test cells P8.1 and P8.2 the high pressure supply system provides fuel at pressures of up to 360 bar. This allows for fundamental research and technology testing of pre-combustion chambers, gas generators and main combustion chambers for rocket propulsion systems. With the launch of test cell P8.3, , compare Fig. 5, a low-pressure supply will also be available, so that investigations on rocket engines with pump supply are possible there.

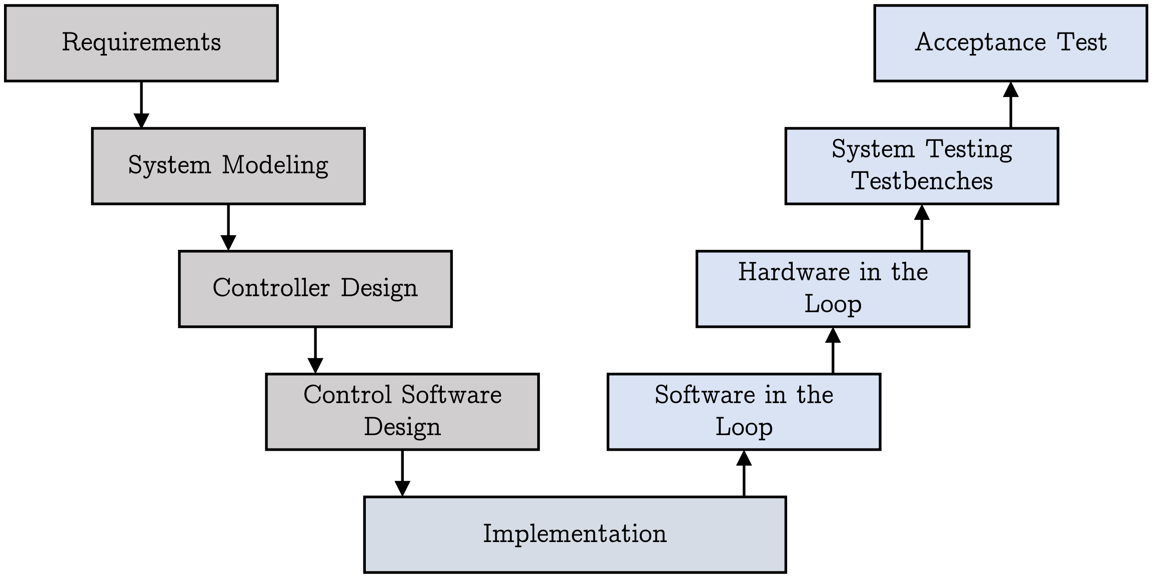


Fig. 6: Development Process

## 6. Zentrum für Telematik (Zft) Test Facility

In order to characterize the thrust vector, the satellite orientation plays a crucial role for resulting orbit changes.[9] The high precision pointing capabilities, which are necessary in a typical dynamic space environment, can be tested with the turntables of "S4 – Smart Small Satellite Systems GmbH" in the ZfT facility. The concept of the Dynamic Bench Test Facility, compare Fig. 7, is based on the combined operation of two high-precision three-axis motion simulators, sensor stimulators and a simulation computer system. Arranging motion simulators and sensor stimulators into proper geometrical configuration enables HIL simulations, as well as material stress tests, sensor characterization, and calibration.

This unique precision facility offers testing of multi-satellite[14] systems with a focus on formation control, relative navigation, inter-satellite communication links and cooperative target observation.[15]

| Axis | Outer | Middle | Inner |
|---|---|---|---|
| Range [°] | ±120 | ±120 | $\infty$ |
| Torque [N m] | 2x4000 | 2x2500 | 95 |
| Max. Acceleration [° s$^{-2}$] | 1000 | 6000 | 2000 |
| Min. Velocity [° s$^{-1}$] | $10^{-4}$ | $10^{-4}$ | $10^{-4}$ |
| Max. Velocity [° s$^{-1}$] | 150 | 150 | 150 |
| Pointing Accuracy [°] | $10^{-4}$ | $10^{-4}$ | $10^{-4}$ |
| Orthogonality [arc sec] | | | $< 5$ |
| Intersection [mm] | | | 0.2 |

Table 1: Characteristics of the Turntable

## 7. Conclusion

As the demands on the rocket propulsion control systems continue to rise, the control software is also becoming increasingly complex. The complex software requires more powerful hardware in turn. Modern embedded system can handle neural networks and diverse model-based approaches. Thus, extensive testing becomes even more important and is inevitable during the development process to guarantee the reliability of the control system. We sketched a development and test roadmap for rocket propulsion systems. First, the testing is completely performed in simulation. Second, step by step some virtual elements are replaced with real hardware (hardware-in-the-loop tests). Third, real embedded systems are tested with real rocket engines using advanced rocket engine test facilities.

### 7.1 *Acknowledgment*

Icons in Fig. 2, Fig. 3, and Fig. 4 made by Freepik, Payungkead, and prettycons from `www.flaticon.com`. The authors would like to thank Tobias Kaiser for valuable discussions concerning space-graded embedded systems.

Fig. 7: Dynamic Test Benches at ZfT

## References


[1] Hiroya Asakawa, Masahiro Tanaka, Takeshi Takaki, and Kazuhiro Higashi. Component tests of a LOX/methane full-expander cycle rocket engine: Electrically actuated valve. In *Proceedings of the 8th European Conference for Aeronautics and Space Sciences*, Madrid, Spain, 2019.

[2] Marko Bacic. On hardware-in-the-loop simulation. In *Proceedings of the 44th IEEE Conference on Decision and Control*, Seville, Spain, 2005. IEEE.

[3] Jean-Noel Chopinet, Francois Lassoudiere, G. Roz, O. Faye, S. Le Gonidec, Patrick Alliot, and Guedron Sylvain. Progress of the development of an all-electric control system of a rocket engine. In *48th AIAA/ASME/SAE/ASEE Joint Propulsion Conference*, Atlanta, Georgia, 2012. American Institute of Aeronautics and Astronautics.

[4] Stephane Colas, Serge Le Gonidec, Philippe Saunois, Martine Ganet, Antoine Remy, and Vincent Leboeuf. A point of view about the control of a reusable engine cluster. In *Proceedings of the 8th European Conference for Aeronautics and Space Sciences (EUCASS)*, Madrid, Spain, 2019.

[5] Jan Deeken and Günther Waxenegger. LUMEN: Engine cycle analysis of an expander-bleed demonstrator engine for test bench operation. In *Deutscher Luft- Und Raumfahrtkongress*, Braunschweig, Germany, 2016.

[6] Kai Dresia, Günther Waxenegger-Wilfing, Jörg Riccius, Jan Deeken, and Michael Oschwald. Numerically Efficient Fatigue Life Prediction of Rocket Combustion Chambers using Artificial Neural Networks. In *Proceedings of the 8th European Conference for Aeronautics and Space Sciences (EUCASS)*, Madrid, Spain, 2019.

[7] J. Hull, D. Ward, and R.R. Zakrzewski. Verification and validation of neural networks for safety-critical applications. In *Proceedings of the 2002 American Control Conference (IEEE Cat. No.CH37301)*, 2002.

[8] Ming Jin and Javad Lavaei. Stability-certified reinforcement learning: A control-theoretic perspective. *arXiv:1810.11505 [cs]*, 2018.

[9] Alexander Kramer, Philipp Bangert, and Klaus Schilling. UWE-4: First Electric Propulsion on a 1U CubeSat - In-Orbit Experiments and Characterization. *Aerospace 2020*, 7, 2020.

[10] George Lentaris, Konstantinos Maragos, Ioannis Stratakos, Lazaros Papadopoulos, Odysseas Papanikolaou, Dimitrios Soudris, Manolis Lourakis, Xenophon Zabulis, David Gonzalez-Arjona, and Gianluca Furano. High-Performance Embedded Computing in Space: Evaluation of Platforms for Vision-Based Navigation. *Journal of Aerospace Information Systems*, 15(4):178–192, 2018.

[11] Mark Nadeski. Bringing machine learning to embedded systems. *Texas Instruments*, 2019.

 

[12] Sergio Pérez-Roca, Julien Marzat, Hélène Piet-Lahanier, Nicolas Langlois, François Farago, Marco Galeotta, and Serge Le Gonidec. A survey of automatic control methods for liquid-propellant rocket engines. *Progress in Aerospace Sciences*, 107:63–84, 2019.

[13] Wolfgang Roth, Günther Schindler, Matthias Zöhrer, Lukas Pfeifenberger, Robert Peharz, Sebastian Tschiatschek, Holger Fröning, Franz Pernkopf, and Zoubin Ghahramani. Resource-Efficient Neural Networks for Embedded Systems. *arXiv:2001.03048 [cs, stat]*, 2020.

[14] Oliver Ruf, Stephan Busch, and Klaus Schilling. Challenges and Novel Approaches for Testing Large Numbers of Small Satellites. In *Proceedings of the 68th International Astronautical Congress*, Adelaide, Australia, 2017.

[15] Oliver Ruf and Klaus Schilling. Automated Attitude Control Verification of Nano-Satellites on System-Level. In *Proceedings of the 33rd AIAA/USU Conference on Small Satellites*, Logan, USA, 2019.

[16] P. Simontacchi, R. Blasi, E. Edeline, S. Sagnier, N. Ravier, A. Espinosa-Ramos, J. Breteau, and Ph. Altenhoefer. PROMETHEUS: Precursor of new low-cost rocket engine family. In *Proceedings of the 8th European Conference for Aeronautics and Space Sciences (EUCASS)*, Madrid, Spain, 2019.

[17] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World. *arXiv:1703.06907 [cs]*, 2017.

[18] Günther Waxenegger-Wilfing, Kai Dresia, Jan Christian Deeken, and Michael Oschwald. Heat Transfer Prediction for Methane in Regenerative Cooling Channels with Neural Networks. *Journal of Thermophysics and Heat Transfer*, 34(2):347–357, 2020.

[19] Günther Waxenegger-Wilfing, Kai Dresia, Jan Christian Deeken, and Michael Oschwald. A Reinforcement Learning Approach for Transient Control of Liquid Rocket Engines. *arXiv:2006.11108 [cs, eess, math, stat]*, 2020.