

Article

Onboard Processing in Satellite Communications Using AI Accelerators

Flor Ortiz ^{*} , Victor Monzon Baeza , Luis M. Garces-Socarras , Juan A. Vásquez-Peralvo ,
Jorge L. Gonzalez , Gianluca Fontanesi , Eva Lagunas , Jorge Querol  and Symeon Chatzinotas 

Interdisciplinary Centre for Security Reliability and Trust (SnT), University of Luxembourg,
L-1855 Luxembourg, Luxembourg

* Correspondence: flor.ortiz@uni.lu

Abstract: Satellite communication (SatCom) systems operations centers currently require high human intervention, which leads to increased operational expenditure (OPEX) and implicit latency in human action that causes degradation in the quality of service (QoS). Consequently, new SatCom systems leverage artificial intelligence and machine learning (AI/ML) to provide higher levels of autonomy and control. Onboard processing for advanced AI/ML algorithms, especially deep learning algorithms, requires an improvement of several magnitudes in computing power compared to what is available with legacy, radiation-tolerant, space-grade processors in space vehicles today. The next generation of onboard AI/ML space processors will likely include a diverse landscape of heterogeneous systems. This manuscript identifies the key requirements for onboard AI/ML processing, defines a reference architecture, evaluates different use case scenarios, and assesses the hardware landscape for current and next-generation space AI processors.

Keywords: satellite communications; artificial intelligence; onboard process; AI chipset



Citation: Ortiz, F.; Monzon Baeza, V.; Garces-Socarras, L.M.; Vásquez-Peralvo, J.A.; Gonzalez, J.L.; Fontanesi, G.; Lagunas, E.; Querol, J.; Chatzinotas, S. Onboard Processing in Satellite Communications Using AI Accelerators. *Aerospace* **2023**, *10*, 101. <https://doi.org/10.3390/aerospace10020101>

Academic Editor: Davide Ferretto

Received: 23 October 2022

Revised: 11 January 2023

Accepted: 13 January 2023

Published: 19 January 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Satellite communication (SatCom) systems today heavily rely on human expertise and manual operations. Control activity of the satellite system requires high human involvement, resulting in increased operational expenditure (OPEX) and implicit latency in human action, leading to quality of service (QoS) degradation. Moreover, human-based decisions are often far from optimal, leading to inefficient system performance [1–3].

In this context, artificial intelligence (AI) has emerged as a promising alternative to cope with a computationally expensive optimization procedure. Recently, with the exponential increase in available data, machine learning (ML) has become a fundamental technology in different areas of wireless communications [4,5]. In particular, ML has already proven to be a helpful tool to accelerate complex optimization procedures for wireless communications in general, and SatCom [1,2,4,6].

The European Space Agency (ESA) opened an AI-related call for SatComs for the first time in 2019 to investigate the applicability of AI techniques in satellite communications. Several potential use cases were shortlisted during these projects, and a preliminary evaluation of a small number of them was carried out to provide guidelines for future research: (i) SATAI—Machine Learning and Artificial Intelligence for Satellite Communications [7] and (ii) MLSAT—Machine Learning and Artificial Intelligence for Satellite Communication [8].

The overall drawbacks identified in both SATAI and MLSAT are as follows:

- A large and representative training set (labeling) is required to achieve acceptable performance.
- Particular use cases were considered, and typically limited ML techniques were examined.
- Insufficient evaluation and comparison with non-ML designs (note that the evaluation phase was < 4 months).

Both activities were too brief for an in-depth analysis of the full potential of ML techniques.

Outside Europe, NASA has been actively investigating the cognitive radio (CR) framework in satellite communications within the John H. Glenn Research Center testbed radios aboard the International Space Station [9].

In addition, Kato et al. [10] propose to use AI techniques to optimize space–air–ground integrated networks (SAGINs). First, the authors discuss several main challenges of SAGINs and explain how AI can solve these problems. They consider satellite traffic balancing and propose a deep learning (DL)-based method to improve traffic control performance. Simulation results conclude that the DL technique can be an effective tool to enhance the performance of SAGINs. However, the authors mention that implementing AI techniques in SAGINs is still a new topic and needs more effort to improve its performance.

There are more significant contributions regarding radio resource management (RRM) in SatCom. For example, in [11], the authors propose a combined learning and optimization approach to address a mixed-integer convex programming (MICP) problem in satellite RRM. Deng et al. [12] suggest an innovative RRM framework for next-generation heterogeneous satellite networks (HSNs), which can cooperate between independent satellite systems and maximize resource utilization. The critical points of the proposed design lie in the architecture that supports intercommunication between different satellite systems and in the management that provides pairing between resources and services. The authors apply deep reinforcement learning (DRL) in the system due to its strong capability for optimal pairing.

Following more recent research suggesting DRL algorithms to solve the RRM problem, Ferreira et al. [13] claim that a feasible solution can be designed for real-time, single-channel resource allocation problems. However, in their study, DRL architectures are based on the discretization of resources before allocation, while satellite resources, such as power, are inherently continuous. Therefore, Luis et al. [14] explore a DRL architecture for power allocation that uses continuous, stateful action spaces, avoiding the need for discretization. However, the policy is not optimal since part of the demand is still lost.

On the other hand, Liu et al. [15] suggest a new dynamic channel allocation algorithm (DRL-DCA) in multibeam satellite systems. The results show that this algorithm could achieve lower blocking probability than traditional algorithms. However, the joint channel and power allocation algorithm is not considered.

Liao et al. [16] construct a game model to learn the optimal strategy in the SatComs scenario. In particular, the authors suggest a bandwidth allocation framework based on the bandwidth DRL, which can dynamically allocate the bandwidth in each beam. The effectiveness of the proposed method in time-varying traffic and large-scale communication is verified in the bandwidth management problem with acceptable computational cost. However, only one resource per satellite can be managed with this method, a critical limitation when full flexibility is sought in the multibeam satellite system.

In [17], a DRL architecture based on a cooperative multiagent system was presented, demonstrating better performance than a single agent for RRM in multibeam satellites with more than one flexibility resource since the number of possible states increases exponentially for a single agent. In addition, Q-learning (QL), deep Q-learning (DQL), and double deep Q-learning (DDQL) algorithms are analyzed by comparing their performance for RRM based on their throughput, complexity, and added latency.

Based on the above, we can note that the study of AI applications in SatCom has advanced considerably in recent years. However, it is still in its infancy; for example, most of these studies are based on theoretical simulations performed on traditional processors without considering the hardware technical limitations in space. AI algorithms can be very computationally intensive, consume significant power, and are slow when running on standard processors. These problems severely limit their use in satellite payloads and, thus, applications. The industry has developed AI-specific processors capable of executing complex algorithms in real time and consuming a fraction of the power. These processors are now available as commercial AI chipsets.

The availability of these chipsets has enabled the practical use of AI for satellite applications in terrestrial and onboard scenarios. However, next-generation defense systems and commercial broadband satellites require more autonomy, onboard data processing, and decision making. In this regard, interest in using commercial AI chipsets has increased significantly to evaluate, develop, and validate AI-based signal processing techniques onboard the satellite, such as signal identification, spectrum monitoring, spectrum sharing, or signal demodulation.

This paper identifies the key performance indicators (KPIs) for AI/ML processing onboard SatCom systems, defines a reference architecture, analyzes different use scenarios, and assesses the hardware landscape for current and next-generation commercial AI processors. The main contributions of the paper are as follows:

- The critical requirements for onboard AI/ML processing include power consumption, latency, and accuracy.
- Usage scenarios for onboard AI/ML processing are identified and discussed, including applications in communication satellites.
- Three scores for comparing the applicability of onboard AI/ML processing are defined, including the onboard applicability, the AI gain, and the complexity. These scores provide a quantitative measure of the suitability of various scenarios where AI chipsets can be used for onboard AI/ML processing.
- The authors present a comprehensive review of commercial AI chipsets and compare their specifications concerning the applicability of onboard AI/ML processing.

Overall, this paper thoroughly analyzes the current state of the art in onboard AI/ML processing, and identifies key considerations for its implementation in communication satellites. It also presents a useful framework for comparing and evaluating the suitability of different AI chipsets for onboard AI/ML processing. The remainder of the paper is divided as follows. In Section 2, we present the KPIs and architecture required for AI processing implementation onboard the satellite. Section 3 evaluates the applicability in different use cases, Section 4 evaluates commercial AI chips for implementation, and Section 5 presents the conclusions and challenges.

2. Onboard AI Processing

AI techniques for onboard processing must go through two phases: training and inference. Initially, the AI model undergoes the training phase. The objective is to find the optimal model parameters to predict the satellite configuration for the system conditions. Then, the trained model is obtained and used to indicate the system parameters depending on the input data. We propose an architecture (Figure 1) based on training the offline ML model with a training database that describes the system behavior. A model that manages the satellite could be obtained according to the system conditions. The main advantage of this architecture is that onboard processing times and required power consumption are reduced, firstly because the model has been previously trained. However, this architecture depends on the training data and the models used.

Not all use cases are implementable onboard the satellite. Therefore, in Table 1, we present the analysis and scoring of the KPIs for the applicability of onboard AI implementation use cases. Each use case should be evaluated accordingly.

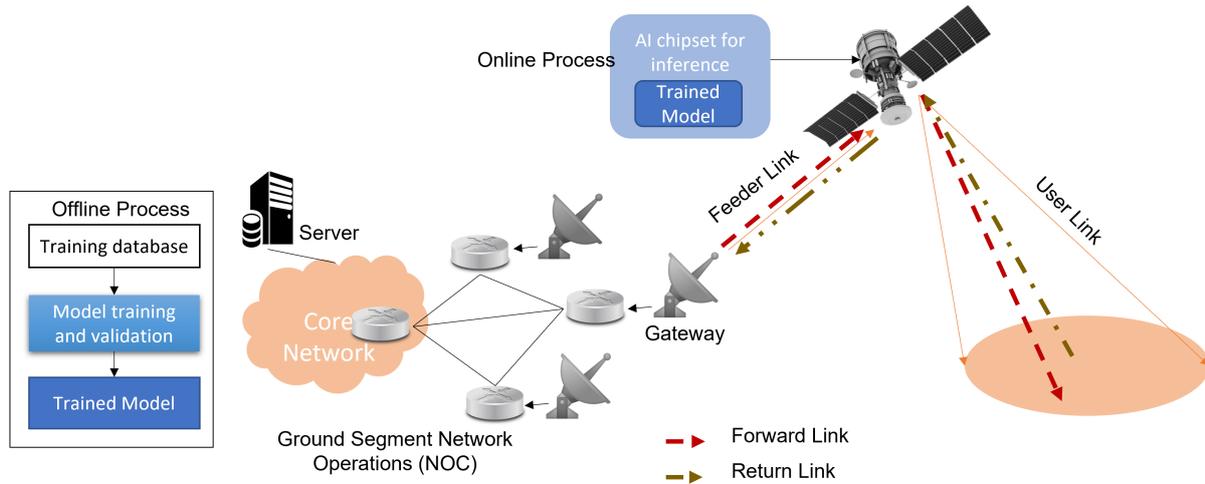


Figure 1. AI chipset implementation in the space segment for onboard processing. The training of the ML algorithm is performed offline in the ground segment with a training database, thus saving the obtained model, and the AI chipset can go on board the satellite using the model for inference.

Table 1. KPI analysis for the applicability of onboard AI implementation use cases.

KPI	Scores					
	0	1	2	3	4	5
Onboard applicability	Not applicable	Network of regenerative satellites	Regenerative Upper layers (CU-RAN)	Regenerative Lower layers (DU-RAN)	Regenerative PHY layer (RU-RAN)	DTP/RF/Antenna Interface
AI Gain	No gain	Human intervention reduction	The previous score + OPEX reduction	The previous score + QoS gain	The previous score + processing time reduction	The previous score + performance gain
Complexity	Unfeasible	Extensive payload changes required	Computational level + memory-intensive + power and time constraints	Computational level + memory-intensive + power constraints	Computational level + memory-intensive	Computational level

2.1. Onboard Applicability

The applicability of the proposed use case to be implemented on board a satellite must be evaluated, considering the state of satellite payloads and their required onboard functionalities. This depends on the current level of technology. Some proposed cases (e.g., adaptive coding and modulation (ACM) optimization) require the onboard implementation of a complete satellite payload or a constellation. We propose a progressive scale of applicability scores that increase according to the complexity of the satellite or the regenerative level required to implement the proposed scenario. These levels can also correspond to the 4G/5G/6G radio access network (RAN) divisions and their functional groups: radio unit (RU), distributed unit (DU), and centralized unit (CU).

2.2. AI Gain

Existing solutions for the proposed use cases often have inherent limitations in translating theory to practice when handling the computational complexity and/or latency required for results, especially when dealing with an ample search space or high degrees of freedom. This has been the typical motivation for using AI, as it has shown a strong potential to overcome this challenge through data-driven solutions. This paper will con-

sider these criteria and evaluate the expected latency, complexity, and throughput gains achieved with AI-based techniques in each considered use case.

2.3. Complexity

When assessing the applicability of the AI technique, the complexity and runtime analysis of the algorithm must always be discussed and taken into account. Consequently, it is crucial to quantify the resources required to execute each selected AI-based technique. This includes the following:

- Computational complexity. Number of operations as a function of the input size of an algorithm.
- Computational time. The execution time of an algorithm depends on the machine that executes it.
- Memory. Memory in neural networks is needed to store input data, weight parameters, and activations as input propagates through the network.
- Power consumption. Power is critical in both phases of AI-based techniques, the training phase and the testing phase. All available processing elements are used in a highly parallel fashion for training. For the inference phase, algorithms are optimized to maximize performance for the specific task for which the model is designed.

3. Onboard AI Applications

3.1. Use Cases

3.1.1. Interference Detection

In a satellite communications system, inter-system interference due to jamming, misaligned dishes, and unintentional interference from other systems has become a significant concern. Interference substantially deteriorates the signal quality, and thus degrades overall system performance. Therefore, detecting interference is the first step in the interference management chain (detection, classification, localization, and mitigation).

The proliferation of the non-geostationary satellite orbit (NGSO) constellation has exacerbated the space interference scenario. With multiple satellites flying in different orbits and inclinations, in-line interference is increasingly likely to occur (see Figure 2) [18].

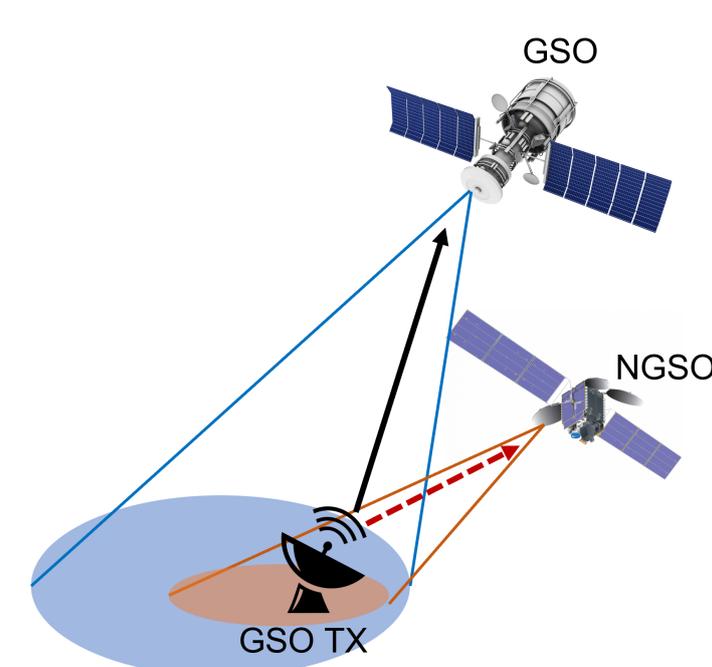


Figure 2. Example of an onboard interference scenario between NGSO and GSO.

Interference is often monitored on the ground and minimized in satellite networks to maintain high QoS. In terms of better spectral management and customer incident prevention, the possibility of offloading a purely human task, such as power spectrum density verification, to an automated system capable of detecting the presence of unwanted signals is an exciting prospect. The introduction of AI would significantly reduce incident/degradation duration and increase cost savings from an OPEX perspective, and ultimately increase availability and QoS.

Significant interference is considered a critical problem for SatCom systems and services. The SatCom industry is increasingly concerned about managing and mitigating interference effectively. Although efficient techniques exist to control substantial interference in SatCom, weak interference is not so easy to detect due to its low signal-to-interference-plus-noise ratio (SINR). To solve this problem, the work proposed in [19] offers and develops a technique that is performed on board the satellite by decoding the desired signal, removing it from the total received signal, and applying an energy detector (ED) on the remaining signal for interference detection. Unlike previous work, the authors in [19] consider imperfect signal cancellation, examining how decoding errors affect detection performance.

The work presented in [20] proposes a two-step algorithm for onboard interference detection, exploiting the frame structure of the digital video broadcasting–second generation satellite extensions (DVB-S2X) standard, which employs pilot symbols for data transmission. Assuming that the pilot signal is known at the receiver, it can be removed from the total received signal. Then, an ED technique can be applied to the remaining signal to decide the presence or absence of interference.

On the other hand, a receiver in which compressive signal processing is used to estimate the power spectrum in the compressed domain using a computationally lightweight algorithm is proposed in [21]. It has a significantly low detection time to detect the presence of a narrowband, high-power interference signal. It also proposes interference filtering using a novel and computationally efficient filtering method in the compressed domain. This receiver architecture results in a computationally efficient receiver with reduced detection time and is suitable for real-time applications. This will help protect a satellite transponder from saturation due to unwanted high-power narrowband interference.

In this context, AI-based techniques can bring gains in identification accuracy and latency. Several AI-based decision processes can be used when mapping baseline features to a class label. Probabilistic-derived decision trees on expert modulation features were among the first to be used in this field, but for many years such decision processes have also been trained directly on datasets represented in their feature space (e.g., neural networks and support vector machines (SVM)).

The accuracy of the interference detector is a critical parameter because false negatives can have a high cost in lowering the SINR, which would be reflected in the QoS. In that sense, using ML for interference detection is expected to decrease the probability of false detection by 44% compared to traditional methods [1].

An ML model based on an autoencoder is proposed for interference detection, assuming the interference is an anomaly in the system. The autoencoder will be composed of an encoder neural network and a decoder neural network, stacked sequentially. The encoder compresses the data blocks and reduces the dimensionality of the input. The data is passed to the decoder, which increases the dimensions and restores the original dimension.

If the input signal does not include any interference, the reconstructed signal will be very close to the input signal. On the contrary, if the input signal comprises interferences that modify the original statistics, the reconstructed signal will be noticeably different (see Figure 3).

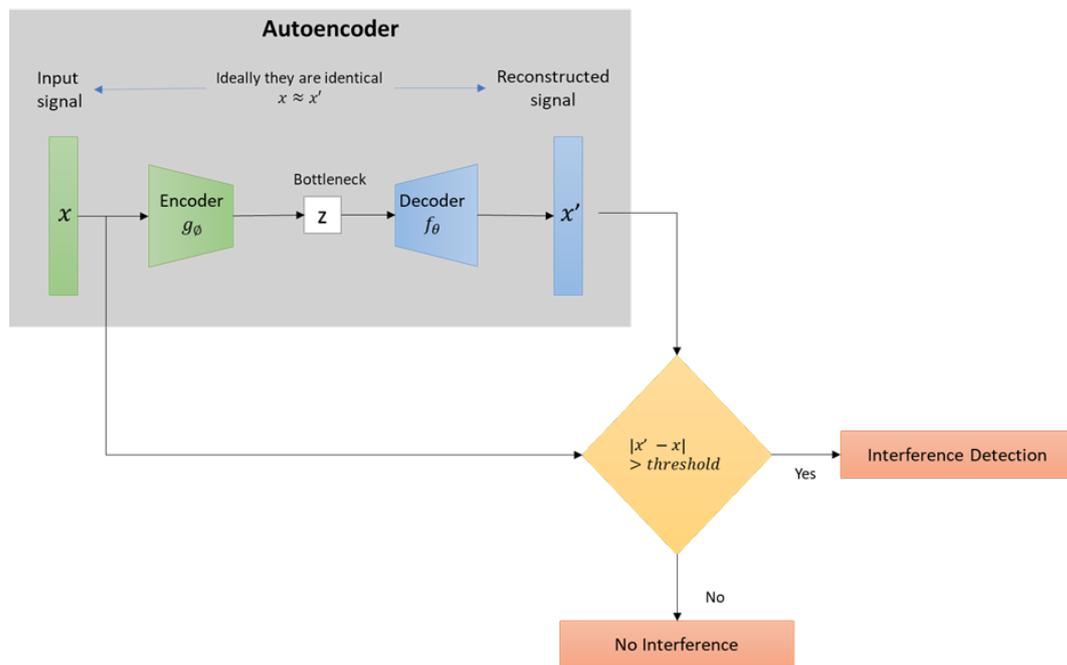


Figure 3. Autoencoder for interference detection.

3.1.2. FEC in Regenerative Payload

Forward error correction (FEC) [22] decoding algorithms such as low-density parity check (LDPC), Reed–Solomon, or polar codes are crucial elements of modern digital communications based on 3GPP and DVB standards. These decoding schemes require high implementation complexity and power consumption to be integrated into full regenerative satellite payloads. Therefore, this use case is motivated by AI acceleration through ML algorithms to reduce the complexity and, thus, the power consumption of FEC decoding algorithms on board satellites. A conventional FEC approach is based on maximum a posteriori decoding in which ML application is restricted to shortcodes due to the exponential training complexity [23]. In addition, sometimes, an unknown distribution of the channel noise makes training difficult. The most popular algorithm applied to conventional FEC decoding with ML is the belief propagation (BP) algorithm [23], trained as a neural network to improve error correction performance. However, its application on regenerative payloads and SatCom links has not been analyzed.

In this scenario, a complexity reduction in FEC decoding for small power-limited regenerative satellites with a reduced number of datalinks is expected. Then, this approach may also be scaled up to future regenerative ultra-high-throughput satellites (UHTSs) with thousands of datalinks requiring simultaneous FEC decoding. A significant decrease in the power consumption of such a case will also lead to an increase in the available power budget for other satellite subsystems or to a decrease in the mass, and hence, launching costs.

Typical decoding relies on maximum a posteriori decoding, which consists of computing the probability that a specific bit is 0 or 1 and selecting the hypothesis with a higher probability. The two main drawbacks of typical decoding are (i) the computational complexity and (ii) the sometimes unknown distribution of the channel noise. Therefore, it makes sense to consider the ML alternative to learning from data automatically. The application of ML to FEC decoding is generally restricted to shortcodes due to the exponential training complexity. For instance, a message with k bits gives a total of 2^k possible codewords. Fortunately, the ACM codes of DVB-S2X are generally limited, favoring such a scenario.

Two ML technique options exist, artificial neural networks (ANNs) and recurrent neural networks (RNNs), as candidates to include onboard processing. The first has a fixed input layer size, which can be inefficient when applying ACM techniques to maintain avail-

ability in the system. The second can process a sequence of arbitrary length by recursively applying a transition function to the internal hidden state vector of the input sequence. The RNN properties suggest that it can correct errors in the time series data flow by referring to the contexts obtained by supervised training. Hence, RNNs have been shown to perform well for FEC decoding on receivers that may not previously know the channel coding types, such as cognitive radio. In this sense, a technique based on RNNs is preferable for error correction based on FEC for a regenerative satellite.

An RNN can process a sequence of arbitrary length by recursively applying a transition function to the internal hidden state vector of the input sequence, in addition to the fact that RNNs have been shown to perform well for FEC coding on receivers that may not know the channel coding types previously, such as cognitive radio [24]. In this sense, a technique based on RNNs is chosen for error correction in a regenerative satellite. The RNN is a direct adaptation of the standard feedforward neural network used to learn the features of time series data. Figure 4 shows the structure of the RNN for the FEC of a regenerative satellite. Hidden layer outputs are used as inputs of the delay layer, and those of the delay layer and the outputs of the delay layer are used as part of the inputs of the hidden layer the next time. Due to this structure, the RNN makes it possible to obtain outputs considering the inputs at a given time and the relationship between its inputs and those of the previous time step. These properties of the RNN suggest that the RNN can correct errors in the time series data flow by referring to the contexts obtained by supervised training. We can use the RNN as an FEC algorithm in the time series data.

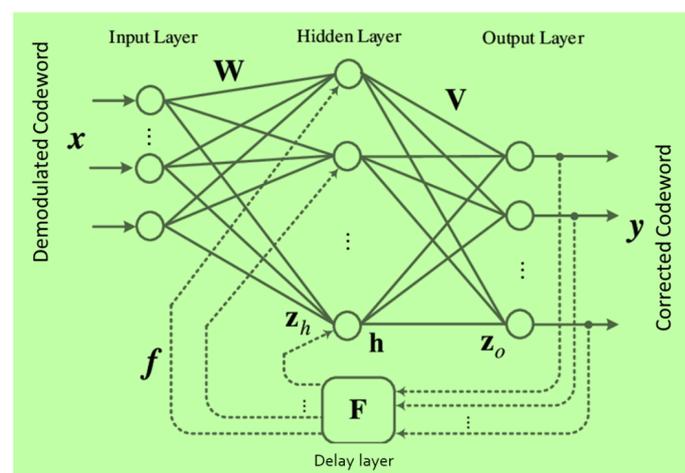


Figure 4. Recurrent neural network for forward error correction in regenerative payloads.

3.1.3. Link Adaption/ACM Optimization

ACM techniques are among the most successful fade mitigation techniques for wireless links. The receiver estimates the instantaneous signal-to-noise ratio (SNR), and signals this value back to the transmitter, requiring a dead time to react to any changes. The channel conditions in SatCom links, such as rain, delay, and scintillation, make it difficult to obtain the estimation signal required for feedback to choose the most suitable MODCOD at all times. In addition, another problem is added, the number of channel models currently considered for NGSO shows no consensus to determine the unique channel model [25]. In conventional approaches, ACM is easy for a single link without interference, where a pilot signal can be used to measure the channel and report back to the receiver. However, the innate delay of satellite links makes the problem substantially challenging. The ML algorithms are expected to improve the channel state information (CSI) prediction in the satellite network to help achieve ACM, hence optimizing system capacity.

The work presented in [26] focuses on applying ML algorithms to CSI prediction in the satellite network, and using the improved prediction results to optimize system capacity. By testing different ML algorithms, the improvement in system performance and

the feasibility of deploying an ML-based prediction framework are demonstrated. The ML-based CSI prediction model provides an average capacity increase of up to 10.9% with acceptable overhead. Faced with the complexity of channel estimation techniques, non-coherent schemes do not need CSI to arise. In SatCom, these schemes are challenging for many use cases; for instance, they require abundant computational resources, as presented in [27] for SatCom. Onboard processing ML techniques can reduce the ACM capability required by non-coherent receivers.

Machine learning techniques are proposed to increase the system’s availability when link adaptation is required. Implementation based on deep learning, such as a long short-term memory network (LSTM), allows us to predict the SINR through time series data. This prediction strategy will enable us to develop a more efficient ACM mechanism.

An LSTM layer is an RNN that supports time series and sequential data. To predict the values of future time steps of a sequence, we can train a sequence-by-sequence regression LSTM network. The responses are the training sequences with the value shifted by a one-time step, i.e., at each time step of the input sequence, the LSTM network learns to predict the value of the next step, $t + 1$.

Figure 5 shows the proposed architecture for SNIR prediction. The SINR time series vector for each n^{th} channel, γ_n , is preprocessed using feature scaling, also known as normalization, which is expressed as γ'_n . The LSTM network comprises at least one LSTM layer and a complete connection layer. The input of the LSTM network is γ'_n , with which the SNIR prediction for time step $t + 1$, $\hat{\gamma}_n$, is performed. With this prediction, the predicted value of the spectral efficiency can be calculated, and the MODCOD can be adapted.

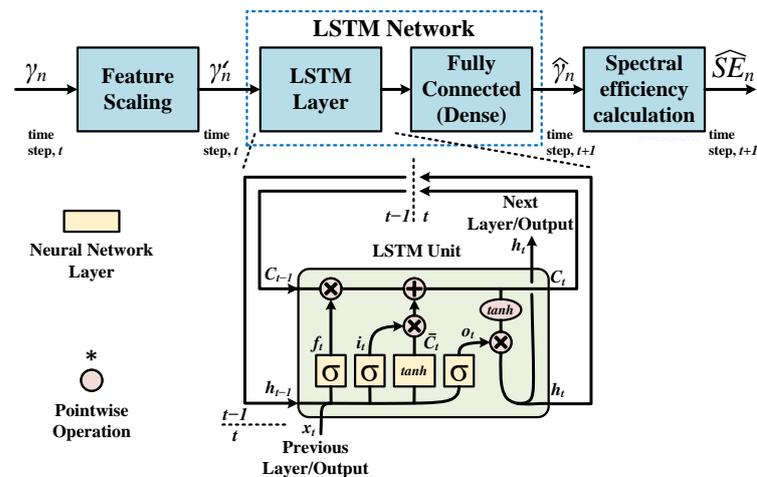


Figure 5. LSTM network for predicted SNR.

3.1.4. Flexible Payload Reconfiguration

Flexible payloads are becoming mainstream, and are revolutionizing the conventional idea of “mission by satellite”. Focusing on frequency-flexible satellites, the goal is to dynamically adapt frequency resources to avoid congestion and ensure a high QoS. More advanced satellite systems can allocate bandwidth dynamically to each beam (assuming a multibeam satellite system). Multiple carriers can be made available within each beam, and the per-user carrier assignment can also be optimized. However, the degree of flexibility available in most satellite systems is power allocation. Power control is easy to implement, and can resolve small congestion events such as the one depicted, where a user terminal is suffering from low SNR (mainly because it is located at the edge of the beam). At the same time, a privileged user in the center of the beam is receiving more SNR than it actually needs. Proper power balancing could resolve the unfortunate situation of the congested user without affecting the QoE of the user in the center of the beam.

The next generation of satellite communications (GSO or NGSO) is being built with the capability to quickly and flexibly assign radio resources according to the system load and

the changing environment. Many significant resource allocation problems are non-convex or combinatorial because of the discrete nature of the variables involved. Hence, computing the optimal solution is challenging, leading to unaffordable computational times.

However, some approaches are based on simplifying the case study to reduce the search variables. For example, the authors in [28] propose an assignment-game-based dynamic power allocation (AG-DPA) to achieve suboptimal low complexity in multibeam satellite systems. The authors compare the results obtained with a proportional power allocation (PPA) algorithm, obtaining a remarkable advantage in energy savings; however, the resource management is still insufficient for the required demand.

As an alternative, Lei et al., have proposed a suboptimal [29] method, which addresses parts of the problem separately and then iteratively adjusts the parameters. The problem splitting is conducted in such a way that power allocation and carrier allocation are separated.

On top of the new perspective of reconfigurable payload, AI shall be integrated into the end-to-end service delivered to introduce a smart interference management system. A fully reconfigurable payload using a conventional interference management approach could not bring a substantial differentiator to the industry or customers. Instead, the winning combination of flexible payload with a smart interference avoidance system is a necessity that can deliver the benefits of the previously mentioned use cases and increase the customer QoS. This use case proposes a machine learning system capable of providing a satellite payload configuration either at the RF interface to meet the requested user bit rates under certain external interference power levels over the coverage area, or at the physical layer (PHY) to provide flexibility for future satellite–terrestrial network integration.

In that sense, it is proposed to use a convolutional neural network (CNN) for flexible payload reconfiguration at the RF interface, as shown in Figure 6. The CNN input is represented in the form of tensor matrices, which will be passed to the convolutional layers where feature extraction is performed and then propagated to the full connection layers, where a classification will be generated, and each class will represent a payload resource configuration. The depth of the input layer will be given by the number of features evaluated as the traffic demand, $r_{i,j}$, the interference power level, $I_{i,j}$, and/or the rain attenuation, $A_{i,j}$, where i and j represent the geographic coordinates within the service area.

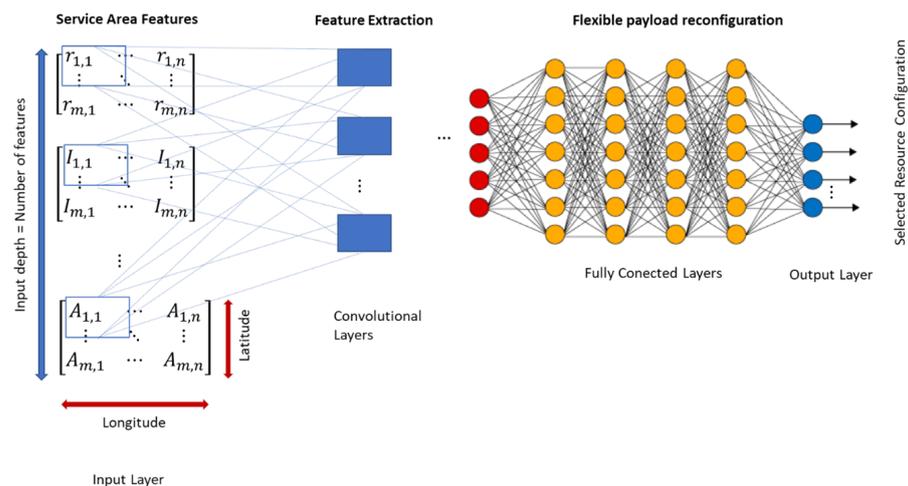


Figure 6. Convolutional neural network for RF interface flexible payload reconfiguration.

3.1.5. Antenna Beamforming

Antenna arrays play an increasingly important role as we move into the era of high-capacity wireless access systems that demand high spectral efficiency. Antenna arrays have become part of the standards for cellular and wireless local area networks. These active antenna arrays will play an equally important role in the next generation of satellite communication systems. In addition, the substantial low earth orbit (LEO) and medium

earth orbit (MEO) constellations envisioned by companies such as OneWeb, Telesat, SES, and SpaceX will require antennas capable of beamforming according to traffic requirements. This convergence of trends is driving a shift from passive antennas with fixed, static beam patterns to active, fully steerable, intelligent antennas. Active beamforming antennas, commonly referred to as active phased array antennas, have active phase shifters on each antenna element or subarray to generate an incremental phase shift to beam steer the radiation pattern in a certain direction. In addition, the amplitude of the power received by each antenna allows to control of the shape of the beam; for instance, control of the beamwidth θ_{-3dB} , side lobe levels (SLLs), and even the nulls in a certain direction. Hence, controlling both the phase and the amplitude gives us plenty of control of the radiation pattern. Next, a short description of the radiation pattern characteristics is provided.

Beamwidth θ_{-3dB}

A direct radiating array (DRA) antenna beamwidth can be increased or decreased by controlling its effective aperture, which also depends on the number of rows and columns of active elements. An estimation of the required number of elements R in one dimension for a certain beamwidth can be computed using Equation (1)

$$R = \frac{\arcsin\left(\frac{1}{\sqrt{2}}\right)}{\eta d \theta_{-3dB}} 2\lambda_0 \quad (1)$$

where η is the antenna efficiency, d is the inter-element spacing, and λ_0 is the antenna wavelength. For instance, to obtain $\theta_{-3dB} = 0.9^\circ$ using a DRA with subarray elements separated by $d = 2.5 \lambda_0$, and assuming an ideal case with $\eta = 1$, a total number of $R = 16 \times 16$ elements is needed. The radiation pattern corresponding to this scenario can be estimated using the planar array factor formula presented in (2)

$$AF = I_{UC} \sum_{r_x=1}^{R_x} e^{j(r_x-1)(kd_x \sin(\theta) \cos(\phi) + \beta_x)} \times \sum_{r_y=1}^{R_y} e^{j(r_y-1)(kd_y \sin(\theta) \sin(\phi) + \beta_y)}, \quad (2)$$

where I_{UC} is the unit cell element pattern, R_x is the number of elements in the x -direction, R_y is the number of elements in the y -direction, k is the wave number, d_x is the period in the x -direction, d_y is the period in the y -direction, β_x is the incremental phase shift in the x -direction, and β_y is the incremental phase shift in the y -direction. The radiation pattern corresponding to the azimuth cut is illustrated in Figure 7.

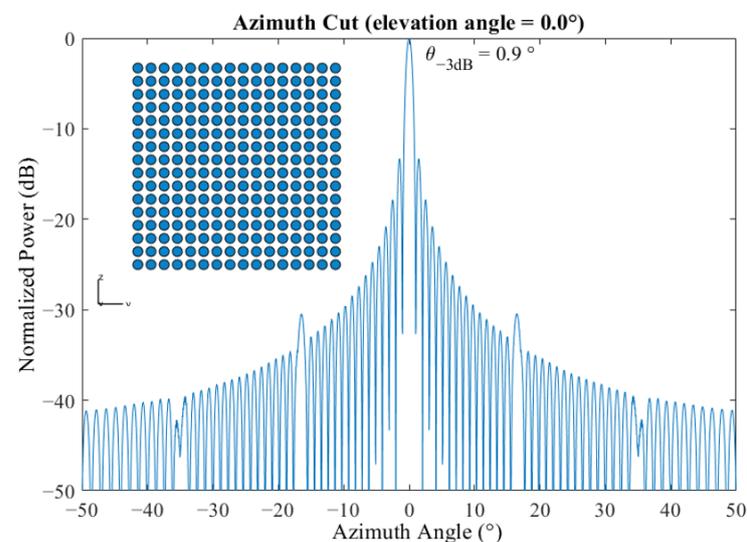


Figure 7. Azimuth cut of a 16×16 DRA, using subarrays as radiating elements separated by $2.5 \lambda_0$.

Side Lobe Levels

Low side lobe levels (SLLs) can be easily accomplished using a windowing operation by means of a filter design, such as that of Hamming, Chebyshev, Kaiser, Taylor, or another known approach. Some of these methods allow one to reduce the side lobe levels to the desired value; however, the drawback of this is a beamwidth increase, hence reducing the directivity of the array. For instance, we can consider the previously presented example, which uses 16×16 subarray elements used to obtain a radiation pattern with a $\theta_{-3dB} = 0.9^\circ$. If we apply a Hamming window to reduce the SLL by 30 dB, we need to increase the number of antenna elements to 24×24 to keep a more or less equal beamwidth. Figure 8 shows the directivity at $\phi = 0^\circ$ cut for the case when the array has tapering for SLL reduction.

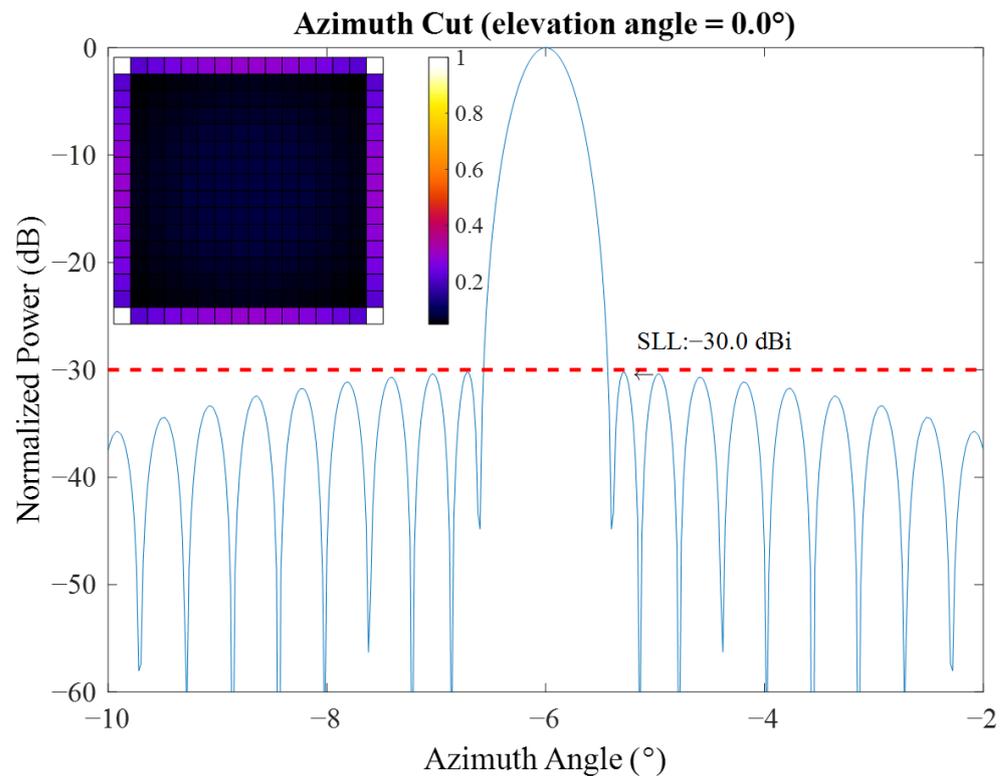


Figure 8. Azimuth cut of a 16×16 DRA using subarrays as radiating elements separated by $2.5 \lambda_0$ with a Hamming window applied to the weight vector.

Nulling

Multibeam interference can be avoided by generating nulls in the radiation pattern at the position of the interfering beam. Controlling the nulling is a straightforward process because the interference direction is known, which makes the implementation of the algorithm relatively simple. First, we need to obtain the weights of a beam that points towards the null angle, and then scale the beam weights and subtract scaled weights from the weights for the beam patterns that point towards the beam steering direction. The formulation of this nulling is presented in (3)

$$W_T = W_{\theta_0\phi_0} - W_{null} \frac{W'_{null} W_{\theta_0\phi_0}}{W'_{null} W_{null}} \tag{3}$$

where $W_{\theta_0\phi_0}$ is the weight vector with the beams steering towards a certain θ_0 and ϕ_0 direction, and W_{null} represents the weight vector with the direction where the null is intended to be. Figure 9 shows the amplitude distribution and radiation pattern of a 16×16 DRA that generates multiple nulling in a multibeam scenario.

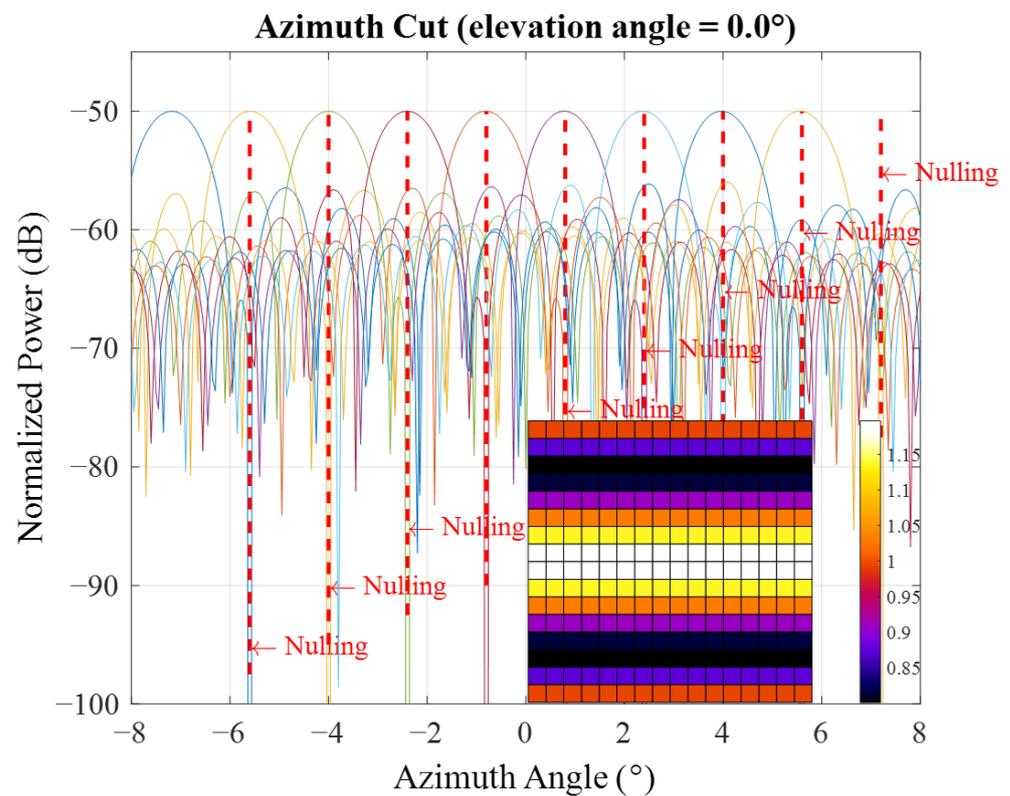


Figure 9. Azimuth cut of 16×16 subarrays in a multibeam scenario with nulling between beams.

All previous parameters can be controlled by AI, especially for a multibeam scenario with SLL, beamwidth, and nulling constraints for multiple beams. Future satellites will be equipped with an array of active antennas rather than a fixed multipoint beam pattern. This allows the generation of multiple spot beams with different numbers, sizes, and shapes.

3.2. Use Cases Assessment

Figure 10 and Table 2 present the analysis and comparison of the different scenarios presented, including the identification of the inputs and outputs of the AI model; the values in the “Total Scores” column represent the sum of each KPI score, as presented in Figure 10. Regarding the applicability on board, the use cases with higher scores are interference detection and antenna beamforming, because the regenerative level remains at RF and antenna level, respectively, as explained in Table 1. Meanwhile, the flexible payload use case is the one that has a lower score concerning onboard applicability because it requires a regenerative payload at the low layer level (DU-RAN) due to the fact that a return channel demodulator would be required to analyze the traffic demand in real time; an additional ML block for onboard traffic prediction is also feasible. Nevertheless, the response time to changing traffic demand is expected to be drastically reduced due to the response time of inference compared to other decision-making techniques.

However, the flexible payload use case is one of those expected to have a higher AI gain due to the complexity of optimal radio resource management with conventional techniques. Therefore, it is expected that with AI-based techniques, an increase in performance of up to 30% and a reduction in power consumption of up to 50% can be obtained [2].

Based on the three KPIs explained in Table 1, the most promising use cases for onboard processing are interference detection, antenna beamforming, and flexible payload.

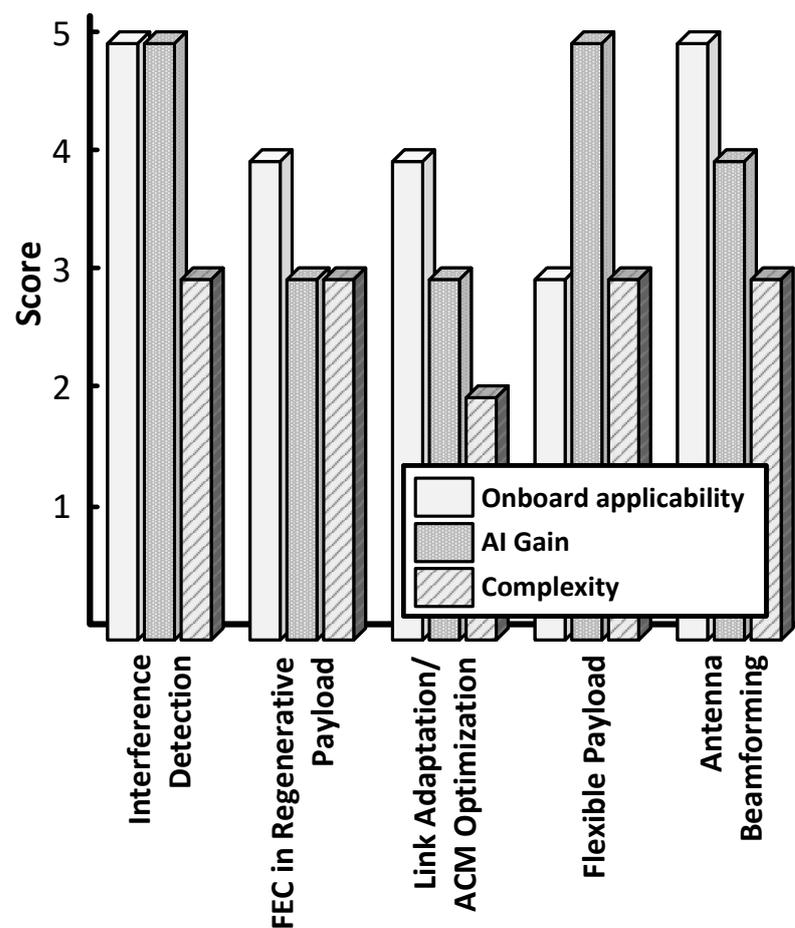


Figure 10. Use cases evaluation.

Table 2. Different scenarios comparison.

Scenario	Input	Output	Total Scores
Interference Detection	Baseband digital signal	Binary detection flag	13
FEC in Regenerative Payload	Modulated or demodulated codeword	Corrected codeword	10
Link Adaptation/ ACM Optimization	SNR time series	Predicted SNR/ModCod	9
Flexible Payload	Demand Information	Configuration of the RF	11
Antenna Beamforming	User locations within antenna FOV	Beamforming parameters	12

4. AI-Capable Commercial Chipsets

So far, complex onboard AI applications can only be performed with very expensive custom-designed ASICs [30]. The increase in performance requirements for onboard processing to support higher data rates and autonomy has made the existing space-graded CPUs obsolete. New technologies, including non-qualified commercial off-the-shelf (COTS) devices from other critical domains, are currently being explored [31].

Finding a device with adequate computer capacity, proper power consumption, and compliance with standards specifications for onboard and standalone applications

is one of the most critical points for future implementations. This section compares non-qualified COTS AI-capable chipsets, and provides guidelines for chip selection.

The current market for commercial AI-capable chipsets is divided into graphics processing units (GPUs), mostly used for the training steps, and dedicated devices for hardware co-processors or stand-alone embedded systems. Focusing on this last classification, several devices have been launched on the market, led mainly by NVIDIA, AMD, Intel, and Qualcomm [32].

Table 3 summarizes current COTS AI-capable chipsets [33–37]. AI-capable chipsets include a central processing unit (CPU) and an on-chip accelerator, which define their capabilities and performance in AI and ML tasks.

Table 3. Trade-off of the AI chipsets and boards under consideration: Core Units.

Device	Provider	Core Units	
		CPU	On-Chip Accelerator
Myriad Family	Intel	2× Leon 4 RISC	Image/Video PA SHAVE
Jetson Nano	NVIDIA	4× ARM Cortex-A57MP	128-CUDA Maxwell
Jetson TX2 Family	NVIDIA	2× Denver ARM64b 4× ARM Cortex-A57MP	256-CUDA Pascal
Jetson Xavier NX Family	NVIDIA	6× Carmel ARM64b	384-CUDA Volta 2× NVDLA 2× PVA 48 Tensor
Jetson AGX Xavier Family	NVIDIA	8× Carmel ARM64b	512-CUDA Volta 2× NVDLA 2× PVA 64 Tensor
Jetson Orin NX Family	NVIDIA	6×/8× ARM64b Cortex-A78AE	1024-CUDA Ampere 1×/2× NVDLAv2 1× PVAv2 32 Tensor
Jetson AGX Orin Family	NVIDIA	8×/12× ARM64b Cortex-A78AE	1792/2048-CUDA Ampere 2× NVDLAv2 1× PVAv2 56/64 Tensor
Cloud AI 100 Family	Qualcomm	Snapdragon 865 MP Kryo 585 CPU	Cloud AI 100
Instinct MI200 Family	AMD	CDNA2	6656/14,080 Stream Proc. 104/220 Core Units
Versal AI Edge Family	XILINX	2× Cortex-A72 2× Cortex-R5F2	8-304 AI Engines 90-1312 DSP Eng 43k-1139k System Logic Cells

The number of CPUs and cores depends on the manufacturer and the family, and they vary from one to two CPUs, and two to twelve cores. Most remarkable are the multicore ARM Cortex-R5F2 on the Versal chips, already on the market, and the 12 cores on the Jetson Orin AGX high-end chip, partially available since late 2022 (according to the information

available on January 2023 [38]). The first is suitable for embedded real-time and safety-critical systems, while the wide number of cores of the second allows the use of multiple threads in sequential parallelizable software [33,37].

In the case of the on-chip accelerators, as their architecture differs from one developer to the other, one cannot make a proper comparison without taking into account the performance per operation, which is detailed in Table 4. The most popular AI on-chip accelerators are GPUs (CUDA Cores, Streams Processors) and AI cores (Tensor, AI Engines, and DSP Engines). Versal chips, apart of the on-chip accelerators (AI and DSP Engines), include a programmable logic (PL) chip that can be used to design time-critical parts of the algorithm, exploiting the inherent parallelism of the hardware design.

Table 4. Trade-off of the AI chipsets and boards under consideration: computing capacity per operation.

Device	Computer Capacity in Operations			Power Cons. (W)
	I8 (Ops)	FP16 (FLOPs)	FP32 (FLOPs)	
Myriad Family	NR	NR	NR	1~2
Jetson Nano	NR	512 G	NR	5–10
Jetson TX2 Family	NR	x	NR	7.5–20
Jetson Xavier NX Family	21T	x	NR	10–20
Jetson AGX Xavier Family	30–32 T	10 T	NR	10–40
Jetson Orin NX Family	70–100 T Sparse	x	x	10–25
	35–50 T Dense			
	20 T Sparse			
Jetson GX Orin Family	108–170 T Sparse	54–85 T	3.3–5.3 T	15–60
	92–105 T Sparse	6.7–10.6 T		
Cloud AI 100 Family	70–400 T	35–200 T	x	15–75
Instinct MI200 Family	181–383 T	181–383 T	45.3–95.7 T (Matrix)	300–560
			22.6–45.9 T	
Versal AI Edge Family	5–202 T	NR	0.4–16.6 T	6–75
	0.6–9.1 T		0.1–2.1 T	
	1–17 T			

Notice that some data are not reported by the developer, marked as *NR*, and in some cases, the required information is not detailed, marked as *x*. The most supported data type operations conducted by the chipsets are integer byte (*INT8*), half-precision floating point (*FP16*), and single-precision floating point (*FP32*) operations. Instinct MI200 and Cloud AI 100 families have the greatest operation rates in half-precision floating point, but their form factor and power consumption make them unsuitable for embedded systems applications. NVIDIA's next-generation AI processors family (Orin NX and AGX Orin) promises to

achieve a high operations per second (*OPs*) rate, as good as the Versal AI chip, increasing the power consumption compared to previous NVIDIA families.

It is important to analyze the computer capacity per Watt in order to select the proper chipset for the application. Figure 11 presents this information for the most common integer bytes operators and floating point.

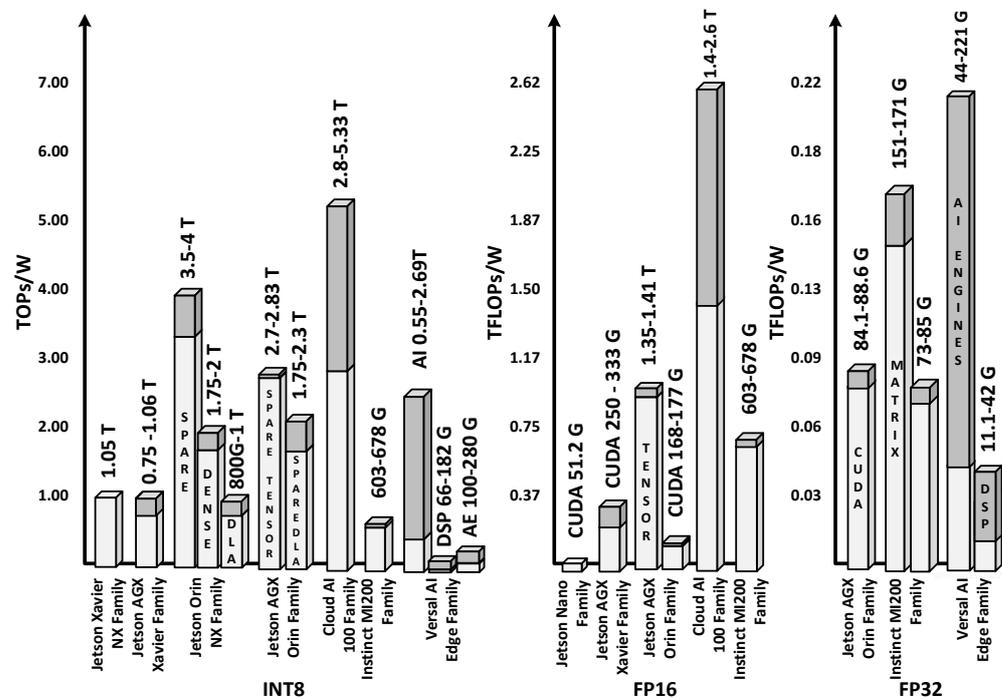


Figure 11. Performance related to the power consumption of the most remarkable families in byte operations (INT8), half-precision floating point operations (FP16), and simple-precision floating point operations (FP32).

The Cloud AI 100 family has better performance regarding byte operations (between 2.8 and 5.33 TOPs/W) followed by the Orin NX family (3.5 and 4 TOPs/W); however, Versal's AI Engines on Edge family exhibits a wide range of performance (0.55 and 2.69 TOPs/W) (INT8). For half-precision floating point (FP16), Cloud AI 100 achieves between 1.4 and 2.6 TFLOPs/W, followed by the Orin AGX family (1.35–1.41 TFLOPs/W); unfortunately, Versal's AI data are not reported for this operation. For single-precision floating point (FP32), Versal's AI Engines achieve the best performance (up to 221 GFLOPs/W), followed by the Instinct MI200 family (between 151 and 171 GFLOPs/W for matrix operations). This comparison proves that Versal AI Edge family has a remarkable and wide range of performance per Watt, being a good solution for AI and ML applications for onboard satellite communications.

Although none of the analyzed COTS AI-capable devices are considered space-graded, some studies are emerging on the use of the COTS chipset in space AI applications. The current literature in this field includes benchmark studies and comparisons of the performance of embedded GPUs (NVIDIA Jetson TX2, NVIDIA Jetson Xavier NX), embedded processors (ARM Cortex-based), and FPGA-based devices (Versal ACAP) on space workload operations [31,39–45]. The principal documented applications are matrix multiplication, useful in CNN implementation [43,44]; onboard infrared detection [31]; onboard space weather detection, including coronal mass ejection (CME) and particle detection [45]; and massive MIMO beamforming for 5G New Radio [46,47].

Working on onboard device requirements, AMD announced the release of the first space-grade XILINX Versal adaptive SoCs, enabling onboard AI processing in space [30,45,48]. The actual information about this device is predicted to be available in early 2023 (according to the information available in January 2023 [30]), and will be

based on the Versal AI Core VC1902, a family not analyzed for its high power consumption (87 mW reported in [49]), including Class B qualification, radiation tolerance, and $45 \times 45 \text{ mm}^2$ packing [48]. There is no information yet about the possible power consumption, the principal inconvenience of its predecessor, and the computer capacity of the chip.

5. Conclusions and Future Challenges

Early work in the literature has demonstrated the progress obtained from using AI in SatCom, yet this research is still in its infancy. Due to the multiple possible use cases, the progress is even more limited when focusing on AI onboard use cases. On the other hand, the interest in evaluating the feasibility of using commercial AI-capable chipsets on board satellites continues to grow. In this sense, we consider different use cases and commercial AI-capable chipsets, and present a trade-off and feasibility analysis for implementing onboard processing. In future work, the analysis should be extended to include a more extensive and detailed study of the use cases and commercial AI chipsets, including a radiation tolerance analysis and a chip comparison based on more recent technologies, such as neuromorphic hardware [50].

Author Contributions: Conceptualization, F.O., E.L. and J.Q.; methodology, F.O. and V.M.B.; validation, E.L., J.Q. and S.C.; formal analysis, F.O., V.M.B., L.M.G.-S. and J.A.V.-P.; investigation, F.O., V.M.B., L.M.G.-S. and J.A.V.-P.; data curation, J.L.G. and G.F.; writing—original draft preparation, F.O., V.M.B., L.M.G.-S., J.A.V.-P., J.L.G. and G.F.; writing—review and editing, J.Q., E.L. and S.C.; supervision, E.L., J.Q. and S.C.; project administration, E.L., J.Q. and S.C.; funding acquisition, E.L., J.Q. and S.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the European Space Agency (ESA) funded under Contract No. 4000134522/21/NL/FGL named “Satellite Signal Processing Techniques using a Commercial Off-The-Shelf AI Chipset (SPAICE)”. Please note that the views of the authors of this paper do not necessarily reflect the views of the ESA. Furthermore, this work was partially supported by the Luxembourg National Research Fund (FNR) under the project SmartSpace (C21/IS/16193290).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ACM	Adaptive Coding and Modulation
AG-DPA	Assignment-Game-based Dynamic Power Allocation
AI	Artificial Intelligence
ANN	Artificial Neural Network
BP	Belief Propagation
CNN	Convolutional Neural Network
COTS	Commercial off-the-Shelf
CPU	Central Processing Units
CR	Cognitive Radio
CSI	Channel State Information
CU	Centralized Unit
DCA	Dynamic Channel Allocation

DDQL	Double Deep Q-Learning
DL	Deep Learning
DQL	Deep Q-Learning
DRA	Direct Radiating Array
DRL	Deep Reinforcement Learning
DSP	Digital Signal Processor
DTP	Digital Transparent Payload
DU	Distributed Unit
DVB-S2X	Digital Video Broadcasting–Second Generation Satellite Extensions
ED	Energy Detector
ESA	European Space Agency
FEC	Forward Error Correction
FOV	Field of Vision
FP	Floating point
GPU	Graphics Processing Unit
GSO	Geostationary Satellite Orbit
HSN	Heterogeneous Satellite Network
KPI	Key Performance Indicators
LDPC	Low-Density Parity Check
LEO	Low Earth Orbit
LSTM	Long Short-Term Memory Network
MEO	Medium Earth Orbit
MICP	Mixed-Integer Convex Programming
ML	Machine Learning
MLSAT	Machine Learning and Artificial Intelligence for Satellite Communication
MODCOM	Modulation and Coding
NGSO	Non-Geostationary Satellite Orbit
NVDLA	NVIDIA Deep Learning Accelerator
OPEX	Operational Expenditure
OPs	Operations per second
PHY	Physical layer
PL	Programmable Logic
PPA	Proportional Power Allocation
PVA	Programmable Vision Accelerator
QL	Q-Learning
QoS	Quality of Service
RAN	Radio Access Network
RNN	Recurrent Neural Network
RRM	Radio Resource Management
RU	Radio Unit
SAGIN	Space–Air–Ground Integrated Networks
SatCom	Satellite Communication
SATAI	Machine Learning and Artificial Intelligence for Satellite Communications
SHAVE	Streaming Hybrid Architecture Vector Engine
SINR	Signal-to-Interference-plus-Noise Ratio
SLL	Side Lobe Level
SNR	Signal-to-Noise Ratio
SVM	Support Vector Machine
UHTS	Ultra-High-Throughput Satellite

References

1. Vazquez, M.A.; Henarejos, P.; Pappalardo, I.; Grechi, E.; Fort, J.; Gil, J.C.; Lancellotti, R.M. Machine Learning for Satellite Communications Operations. *IEEE Commun. Mag.* **2021**, *59*, 22–27. [[CrossRef](#)]
2. Ortiz-Gomez, F.G.; Lei, L.; Lagunas, E.; Martinez, R.; Tarchi, D.; Querol, J.; Salas-Natera, M.A.; Chatzinotas, S. Machine Learning for Radio Resource Management in Multibeam GEO Satellite Systems. *Electronics* **2022**, *11*, 992. [[CrossRef](#)]
3. Kodheli, O.; Lagunas, E.; Maturo, N.; Sharma, S.K.; Shankar, B.; Montoya, J.F.M.; Duncan, J.C.M.; Spano, D.; Chatzinotas, S.; Kisseleff, S.; et al. Satellite Communications in the New Space Era: A Survey and Future Challenges. *IEEE Commun. Surv. Tutorials* **2021**, *23*, 70–109. [[CrossRef](#)]
4. Morocho-Cayamcela, M.E.; Lee, H.; Lim, W. Machine learning for 5G/B5G mobile and wireless communications: Potential, limitations, and future directions. *IEEE Access* **2019**, *7*, 137184–137206. [[CrossRef](#)]
5. Jiang, C.; Zhang, H.; Ren, Y.; Han, Z.; Chen, K.C.; Hanzo, L. Machine Learning Paradigms for Next-Generation Wireless Networks. *IEEE Wirel. Commun.* **2017**, *24*, 98–105. [[CrossRef](#)]
6. Cornejo, A.; Landeros-Ayala, S.; Matias, J.M.; Ortiz-Gomez, F.; Martinez, R.; Salas-Natera, M. Method of Rain Attenuation Prediction Based on Long–Short Term Memory Network. *Neural Process. Lett.* **2022**, 1–37. [[CrossRef](#)]
7. SATAI | ESA TIA. Available online: <https://artes.esa.int/projects/satai> (accessed on 15 September 2022).
8. MLSAT | ESA TIA. Available online: <https://artes.esa.int/projects/mlsat> (accessed on 15 September 2022).
9. Glenn Research Center | NASA. *Cognitive Communications*; Glenn Research Center | NASA: Cleveland, OH, USA **2020**.
10. Kato, N.; Fadlullah, Z.M.; Tang, F.; Mao, B.; Tani, S.; Okamura, A.; Liu, J. Optimizing Space-Air-Ground Integrated Networks by Artificial Intelligence. *IEEE Wirel. Commun.* **2019**, *26*, 140–147. [[CrossRef](#)]
11. Wang, A.; Lei, L.; Lagunas, E.; Chatzinotas, S.; Ottersten, B. Dual-DNN Assisted Optimization for Efficient Resource Scheduling in NOMA-Enabled Satellite Systems. In Proceedings of the IEEE GLOBECOM 2021, Madrid, Spain, 7–11 December 2021.
12. Deng, B.; Jiang, C.; Yao, H.; Guo, S.; Zhao, S. The Next Generation Heterogeneous Satellite Communication Networks: Integration of Resource Management and Deep Reinforcement Learning. *IEEE Wirel. Commun.* **2020**, *27*, 105–111. [[CrossRef](#)]
13. Ferreira, P.V.R.; Paffenroth, R.; Wyglinski, A.M.; Hackett, T.M.; Bilen, S.G.; Reinhart, R.C.; Mortensen, D.J. Multiobjective Reinforcement Learning for Cognitive Satellite Communications Using Deep Neural Network Ensembles. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 1030–1041. [[CrossRef](#)]
14. Luis, J.J.G.; Guerster, M.; Del Portillo, I.; Crawley, E.; Cameron, B. Deep reinforcement learning for continuous power allocation in flexible high throughput satellites. In Proceedings of the IEEE Cognitive Communications for Aerospace Applications Workshop, CCAAW 2019, Cleveland, OH, USA, 25–26 June 2019. [[CrossRef](#)]
15. Liu, S.; Hu, X.; Wang, W. Deep Reinforcement Learning Based Dynamic Channel Allocation Algorithm in Multibeam Satellite Systems. *IEEE Access* **2018**, *6*, 15733–15742. [[CrossRef](#)]
16. Liao, X.; Hu, X.; Liu, Z.; Ma, S.; Xu, L.; Li, X.; Wang, W.; Ghannouchi, F.M. Distributed intelligence: A verification for multi-agent DRL-based multibeam satellite resource allocation. *IEEE Commun. Lett.* **2020**, *24*, 2785–2789. [[CrossRef](#)]
17. Ortiz-Gomez, F.G.; Tarchi, D.; Martinez, R.; Vanelli-Coralli, A.; Salas-Natera, M.A.; Landeros-Ayala, S. Cooperative Multi-Agent Deep Reinforcement Learning for Resource Management in Full Flexible VHTS Systems. *IEEE Trans. Cogn. Commun. Netw.* **2021**, *8*, 335–349. [[CrossRef](#)]
18. Jalali, M.; Ortiz, F.; Lagunas, E.; Kisseleff, S.; Emiliani, L.; Chatzinotas, S. Radio Regulation Compliance of NGSO Constellations' Interference towards GSO Ground Stations. In Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications PIMRC, Kyoto, Japan, 12–15 September 2022.
19. Politis, C.; Maleki, S.; Tsinos, C.; Chatzinotas, S.; Ottersten, B. Weak interference detection with signal cancellation in satellite communications. In Proceedings of the ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing—Proceedings, New Orleans, LA, USA, 5–9 March 2017; pp. 6289–6293. [[CrossRef](#)]
20. Politis, C.; Maleki, S.; Tsinos, C.; Chatzinotas, S.; Ottersten, B. On-board the satellite interference detection with imperfect signal cancellation. In Proceedings of the IEEE Workshop on Signal Processing Advances in Wireless Communications, SPAWC, Edinburgh, UK, 3–6 July 2016. [[CrossRef](#)]
21. Prakash, C.; Bhimani, D.; Chakka, V.K. Interference detection & filtering in satellite transponder. In Proceedings of the International Conference on Communication and Signal Processing, ICCSP 2014—Proceedings, Melmaruvathur, India, 3–5 April 2014; pp. 1394–1399. [[CrossRef](#)]
22. Regenerative Payload Using End-to-End Fec Protection. 2018. Available online: <https://data.epo.org/publication-server/document?iDocId=5548983&iFormat=0> (accessed on 21 October 2022).
23. Nachmani, E.; Be'Ery, Y.; Burshtein, D. Learning to decode linear codes using deep learning. In Proceedings of the 54th Annual Allerton Conference on Communication, Control, and Computing, Monticello, IL, USA, 27–30 September 2016; pp. 341–346.
24. Mei, F.; Chen, H.; Lei, Y. Blind Recognition of Forward Error Correction Codes Based on Recurrent Neural Network. *Sensors* **2021**, *21*, 3884. [[CrossRef](#)] [[PubMed](#)]
25. Monzon Baeza, V.; Lagunas, E.; Al-Hraishawi, H.; Chatzinotas, S. An Overview of Channel Models for NGSO Satellites. In Proceedings of the IEEE Vehicular Technology Conference, VTC-Fall, Beijing, China; London, UK, 26–29 September 2022.
26. Wang, X.; Li, H.; Wu, Q. Optimizing Adaptive Coding and Modulation for Satellite Network with ML-based CSI Prediction. In Proceedings of the IEEE Wireless Communications and Networking Conference, WCNC, Marrakesh, Morocco, 15–18 April 2019. [[CrossRef](#)]

27. Monzon Baeza, V.; Ha, V.N.; Querol, J.; Chatzinotas, S. Non-Coherent Massive MIMO Integration in Satellite Communication. In Proceedings of the 39th International Communications Satellite Systems Conference (ICSSC 2022), Stresa, Italy, 18–21 October 2022.
28. Liu, S.; Fan, Y.; Hu, Y.; Wang, D.; Liu, L.; Gao, L. AG-DPA: Assignment game-based dynamic power allocation in multibeam satellite systems. *Int. J. Satell. Commun. Netw.* **2020**, *38*, 74–83. [[CrossRef](#)]
29. Lei, L.; Lagunas, E.; Yuan, Y.; Kibria, M.G.; Chatzinotas, S.; Ottersten, B. Beam Illumination Pattern Design in Satellite Networks: Learning and Optimization for Efficient Beam Hopping. *IEEE Access* **2020**, *8*, 136655–136667. [[CrossRef](#)]
30. AMD Inc. *First Space-Grade Versal AI Core Devices to Ship Early 2023*; AMD Inc.: Santa Clara, CA, USA, 2022.
31. Rodriguez, I.; Kosmidis, L.; Notebaert, O.; Cazorla, F.J.; Steenari, D. An On-board Algorithm Implementation on an Embedded GPU: A Space Case Study. In Proceedings of the Design, Automation and Test in Europe Conference and Exhibition, DATE 2020, Grenoble, France, 9–13 March 2020; Institute of Electrical and Electronics Engineers Inc.: Grenoble, France, 2020; pp. 1718–1719. [[CrossRef](#)]
32. Pang, G. The AI Chip Race. *IEEE Intell. Syst.* **2022**, *37*, 111–112. [[CrossRef](#)]
33. NVIDIA. Jetson Modules. 2022. Available online: <https://developer.nvidia.com/embedded/jetson-modules> (accessed on 14 September 2022).
34. AMD. AMD Instinct™ MI250 Accelerator. 2021. Available online: <https://www.amd.com/en/products/server-accelerators/instinct-mi250> (accessed on 14 September 2022).
35. Intel. Intel® Movidius™ Myriad™ X Vision Processing Unit 4GB. 2019. Available online: <https://ark.intel.com/content/www/us/en/ark/products/125926/intel-movidius-myriad-x-vision-processing-unit-4gb.html> (accessed on 15 September 2022).
36. Qualcomm. *Qualcomm® Cloud AI 100*; Product Description; Qualcomm Technologies, Inc.: San Diego, CA, USA, 2019.
37. Xilinx. *Versal™ AI Core Series Product Selection Guide*; User Guide; Advanced Micro Devices, Inc.: Santa Clara, CA, USA, 2021.
38. NVIDIA Corp. *NVIDIA Jetson Roadmap*; NVIDIA Corp.: Santa Clara, CA, USA, 2022.
39. Kosmidis, L.; Lachaize, J.; Abella, J.; Notebaert, O.; Cazorla, F.J.; Steenari, D. GPU4S: Embedded GPUs in Space. In Proceedings of the 22nd Euromicro Conference on Digital System Design, DSD 2019, Kallithea, Greece, 28–30 August 2019; Institute of Electrical and Electronics Engineers Inc.: Kallithea, Greece, 2019; pp. 399–405. [[CrossRef](#)]
40. Kosmidis, L.; Rodriguez, I.; Jover, Á.; Alcaide, S.; Lachaize, J.; Abella, J.; Notebaert, O.; Cazorla, F.J.; Steenari, D. GPU4S: Embedded GPUs in space - Latest project updates. *Microprocess. Microsystems* **2020**, *77*, 103143. [[CrossRef](#)]
41. Kosmidis, L.; Rodriguez, I.; Jover, Á.; Alcaide, S.; Lachaize, J.; Abella, J.; Notebaert, O.; Cazorla, F.J.; Steenari, D. GPU4S (GPUs for Space): Are we there yet? In Proceedings of the European Workshop on On-Board Data Processing (OBDP), Online, 14–17 June 2021; pp. 1–8.
42. Steenari, D.; Forster, K.; O’Callaghan, D.; Tali, M.; Hay, C.; Cebecauer, M.; Ireland, M.; McBerren, S.; Camarero, R. Survey of High-Performance Processors and FPGAs for On-Board Processing and Machine Learning Applications. In Proceedings of the European Workshop on On-Board Data Processing (OBDP), Online, 14–17 June 2021; p. 28.
43. Rodriguez, I.; Kosmidis, L.; Lachaize, J.; Notebaert, O.; Steenari, D. *Design and Implementation of an Open GPU Benchmarking Suite for Space Payload Processing*; Universitat Politècnica de Catalunya: Barcelona, Spain, 2019; pp. 1–6.
44. Steenari, D.; Kosmidis, L.; Rodriguez-Ferrandez, I.; Jover-Alvarez, A.; Forster, K. OBPMARK (On-Board Processing Benchmarks) - Open Source Computational Performance Benchmarks for Space Applications. In Proceedings of the European Workshop on On-Board Data Processing (OBDP), Online, 14–17 June 2021; pp. 14–17.
45. Marques, H.; Foerster, K.; Bargholz, M.; Tali, M.; Mansilla, L.; Steenari, D. Development methods and deployment of machine learning model inference for two Space Weather on-board analysis applications on several embedded systems. In Proceedings of the ESA Workshop on Avionics, Data, Control and Software Systems (ADCSS), ESA, Virtual, 16–18 November 2021; pp. 1–14.
46. AMD XILINX Inc. *5G Beamforming with Versal AI Core Series*; Technical Report; AMD XILINX Inc.: San Jose, CA, USA, 2021.
47. XILINX Inc. *Beamforming Implementation on AI Engine*; Technical Report; XILINX Inc.: San Jose, CA, USA, 2021.
48. AMD XILINX Inc. *XQR Versal for Space 2.0 Applications*; Product Brief; AMD XILINX Inc.: San Jose, CA, USA, 2022.
49. AMD XILINX Inc. *AI Inference with Versal AI Core Series*; Technical Report; AMD XILINX Inc.: San Jose, CA, USA, 2022.
50. Ortiz, F.; Lagunas, E.; Martins, W.; Dinh, T.; Skatchkovsky, N.; Simeone, O.; Rajendran, B.; Navarro, T.; Chatzinotas, S. Towards the Application of Neuromorphic Computing to Satellite Communications. In Proceedings of the 39th International Communications Satellite Systems Conference (ICSSC), Stresa, Italy, 18–21 October 2022.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.