DATA SCIENCE AND ARTIFICIAL INTELLIGENCE FOR COMMUNICATIONS

Machine Learning for Satellite Communications Operations

Miguel Ángel Vázquez, Pol Henarejos, Irene Pappalardo, Elena Grechi, Joan Fort, Juan Carlos Gil, and Rocco Michele Lancellotti

ABSTRACT

This article introduces the application of machine learning (ML)-based procedures in real-world satellite communication operations. While the application of ML in image processing has led to unprecedented advantages in new services and products, the application of ML in wireless systems is still in its infancy. In particular, this article focuses on the introduction of ML-based mechanisms in satellite network operation centers such as interference detection, flexible payload configuration, and congestion prediction. Three different use cases are described, and the proposed ML models are introduced. All the models have been constructed using real data and considering current operations. As reported in the numerical results, the proposed ML-based techniques show good numerical performance: the interference detector presents a false detection probability decrease of 44 percent, the flexible payload optimizer reduces the unmet capacity by 32 percent, and the traffic predictor reduces the prediction error by 10 percent compared to other approaches. In light of these results, the proposed techniques are useful in the process of automating satellite communication systems.

INTRODUCTION

It may come as a surprise to realize that nowadays, satellite communications still heavily depend on human expertise and manual operations. Satellite operators’ network operations centers require strong human involvement, leading to high operational expenditure (OPEX) and implicit latency in human action that causes degradation of quality of service (QoS). Indeed, ticket processing of incidents in the radio frequency (RF) plane requires the intervention of human experts to provide a technical solution.

OPEX reduction will guide ground satellite segment system equipment. In this context, procedure automation of satellite communications operations will be a key element of future systems. As a matter of fact, automation requires intelligent systems able to process certain inputs from satellite data and translate them into actions or other data. There is a myriad of options for building these systems like knowledge-based ones where rules are conceived considering domain-expert knowledge of the system [1].

However, motivated by the current data gathering processes of satellite systems and the explosion of cloud services for managing this data efficiently, in this article we resort to data-driven techniques. Precisely, we consider the use of deep learning [2]. Deep learning is a field of machine learning (ML) that considers the optimization of cascaded linear and nonlinear operations. Furthermore, deep learning presents tremendous potential whenever a large-scale amount of data can be used.

In this article, three main current procedures of satellite communications operations are revisited; namely, interference detection, flexible payload configuration, and congestion prediction. While interference detection requires raw in-phase and in-quadrature (IQ) samples of a transponder, congestion prediction is based on user rate demand historical data. Furthermore, flexible payload configuration considers both a theoretical model of the payload and certain rate demands over the coverage area.

The application of deep learning in the different use cases is introduced, and the performance of the resulting system is described. For each procedure, we show the benefits of automation based on the new ML-based techniques. Indeed, the proposed data-driven techniques result in a potential substantial reduction of the operational costs of the control center. This cost trimming comes from the assistance information provided by the ML model, which helps network operation center engineers in the decision making process, leading to a reduction of the event processing time. Note that quantitative cost analysis is out of the scope of this article, and the proposed ML models could potentially lead to cost reduction, but this is still speculation.

To the best of the authors’ knowledge, this is the first time the application of ML in commercial satellite operation systems is introduced. Remarkably, other authors have addressed the problem of using ML in link-to-link design of scientific missions [3].

The remainder of the article is as follows. We describe interference detection, flexible payload...
configuration, and congestion prediction use cases and their ML modeling. We then conclude the article.

**Interference Detection**

**Current Interference Management in Satellite Control Centers**

To maintain high QoS and user experience, interference is closely monitored and minimized in satellite networks. Interference detection is typically a task performed on a reactive instead of proactive mode. Given that in most cases satellites simply relay signals coming from the Earth, interfering signals are present to a large extent in all frequency bands. The possibility to offload a purely human task such as power spectrum density check to an automated system, able to detect the presence of unwanted signals, is an exciting perspective in terms of improved spectral management and customer incident avoidance.

Most interference present today is caused by human errors, due to either mispointed antenna (cross-polarization or adjacent satellite) or misconfigured equipment (noise introduction, intermodulation, etc.). These parameters cover 70–80 percent of all interference cases and are not related to terrestrial networks. If there are carrier overlaps, it implies a digital video broadcasting (DVB) carrier overlapping another DVB carrier or a satellite modem transmission, including very small aperture terminal (VSAT) traffic in time-division multiple access (TDMA), for instance.

In order to mitigate them, there are several techniques that may help to reduce the levels of interference. However, on many occasions, it is still difficult to cope with them. Currently, the only way to manage interference is by human intervention and performing an exhaustive analysis, which may take several days to solve the incident. In other words, there are qualified personnel dedicated to detecting interference by inspecting figures, such as the spectrum, abnormal error rate increase, or degraded user experience.

**Autoencoding for Interference Detection**

For the aforementioned interference detection problem, we rely on an unsupervised ML model called an autoencoder [4]. This technique allows the inputs to be reproduced based on previous training. If the inputs are similar to the training dataset, the produced output maintains the same statistics. However, if the inputs are significantly different from the training, the produced output’s statistics are completely different compared to the original. Hence, it is possible to detect signal perturbations that modify the original statistics by measuring the error between the input and output signals.

For our case, the autoencoder is composed by an encoding convolutional neural network (CNN) and a decoding CNN, 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. The activation layer uses the hyperbolic tangent (tanh) function.

**FIGURE 1.** Autoencoder using convolutional neural network, which is employed for interference detection. The triplets are the dimension of data between each layer.

ANN (Autoencoder Neural Network) is composed by several convolutional and decimating layers placed sequentially, as depicted in Fig. 1. A similar structure can also be used for decoding terrestrial signals [5].

If the input signal does not contain any interference — and the statistics are not modified — the reconstructed signal is very close to the input. On the contrary, if the input signal contains interference — and the statistics are not modified — the reconstructed signal is notably different. Hence, at the output of the model we compute the mean squared error (MSE) between the input and the output. This process is repeated for each layer of the captured signal.

The set of MSE samples constitutes a signal whose values are always positive, and decrease if the input does not contain interference and increase if the input contains interference. Moreover, this set also displays properties depending on the input’s statistics. Thus, if the input’s statistics are modified, this is reflected in the statistics of the MSE. This is exploited by our approach.

As scoring methods, we resort to different features of the resulting error; namely, sum of normalized moment deviation, $S_1$, and the Pearson product-moment correlation coefficient between the MSE probability density function (PDF) with and without interference, $S_2$ [6].

To trigger the presence of interference, we use the previous scores to set the different thresholds, $T_1$ and $T_2$. The first score, $S_1$, defines the deviation between the statistics orders to the original. We analyze different datasets, with and without interference, and the results of the different scores. These datasets consist of several captures of different events (cross-polar interference, adjacent satellite interference, etc.) whose MSE out-

IEEE Communications Magazine • February 2021
put error is illustrated in Fig. 2 with the MSE for the case when we do not have any interference. As expected, the resulting PDF is different for almost every event. By simple visual inspection, it is possible to advise the satellite operator that the signal is interfered with at some location. The obtained performance of true positives is 81 percent, whereas the remaining 19 percent is caused by false positives. It is important to remark that scoring values can be adjusted to modify these results. To our understanding, in the interference detection case, it is preferable to minimize false negatives at the expense of false positives.

Based on our experiments, we compared our proposed solution with existing energy detectors. On one hand, in the case where the interfering signal falls in the same band as the desired signal and its energy is lower than the energy of the desired signal, the energy detector is not able to detect any interference and becomes useless. On the other hand, our proposed solution is able to detect the majority of interference. In our experiments, the energy detector throws 34 percent of false positives.

**Flexible Payload Optimization**

**Human-Based Approach**

Future satellites equipped with flexible payloads will allow their resources to be allocated in response to the temporal variations of the scenario such as dynamic traffic demands and undesired interference events. The satellite resources to be configured are the beam pattern, the transmit power, and the frequency allocation. To the best of the authors’ knowledge, this configuration is expected to be performed by an operator using a graphical tool. That is, an operator will provide the Earth coordinates in order to modify the coverage of the satellite. Regarding the frequency and power allocation, the same idea is followed. Bearing this in mind, in the presence of an event, the operator shall manually compute the best resulting payload configuration.

The idea behind the conceived ML model is to assist and eventually eliminate the human intervention in the payload reconfiguration. This will reduce the OPEX and the time to react to system events, leading to an increase of customer QoS. As a general statement, the resulting ML model aims to provide a relevant payload configuration, which is able to assist the operator in producing a new payload configuration.

**Optimization Assisted via Deep Learning**

The return link of a geostationary multibeam satellite system operating at Ku band in the presence of undesired interference is considered here. The satellite attends a total of $N_u$ fixed satellite terminals. Each user terminal generates a traffic request of $R_i$ b/s for $i = 1, \ldots, N_u$. Let $h_i$ be the gain experienced by the transmitted signal of the $i$th user terminal and received by the ground station.

For the return link, the satellite has a set of $N_{ch}$ subcarriers available. We consider that each user terminal can only be served by a subset of frequency subcarriers (i.e., two different user terminals cannot use the same subcarrier). The return link transmission takes place in the presence of an external interference at a certain geographical location. We consider that the interference transmit power is $P_{int}$, and its equivalent channel is represented by $h$. The interference location is assumed known, and the system designer can modify the satellite footprint in order to reject the received interference by a factor $\mu$. This factor is assumed to be $-6, -10, -15, \text{ and } -18$ dB. The interference bandwidth occupation is assumed to be a subset of the $N_{ch}$ channels. As an example, in Fig. 3 we show different beampatterns for a $\mu = -15$ dB in a certain interference location.

The goal is to allocate resources such that each user terminal is offered a capacity ($R$) that is as close as possible to the requested capacity, $R_i$. In particular, we consider the optimization of the unmet capacity (UC) defined as the quadratic average difference between the requested capacity and the offered capacity. To sum up, the conceived model shall produce a mapping between user terminals and subcarriers given a scenario of

![FIGURE 2. Probability density function of the mean square error between the input and output signals employed in the autoencoder. It is observable that different error statistics are obtained for each interference scenario. PDF and MSE values correspond to normalized RF signals and probability.](image)

![FIGURE 3. The footprint refers to a GEO satellite operating at Ku band where interference located in the Middle-East is mitigated \(\mu = -15\) dB. The different contours show the minimum gain offered by the satellite. In particular, the red contour shows \(1.3\) dB coverage, green shows \(-8.7\) dB coverage, and blue shows \(-13.1\) dB coverage.](image)
the DNN rather than pure randomness. We obtain better results by using initial solutions from heuristic optimizations, while the GA assisted via a DNN is able to find better solutions, leading to a huge search space dimension.

In order to reduce the computational complexity of the model, we consider the approach of assisting the GA with certain knowledge learned by a deep neural network (DNN). This philosophy, reported in [8], has been utilized in other optimization techniques such as [9]. Here, our approach is to create a DNN able to generate efficient initial configurations so that the GA can provide efficient solutions in a short time period.

The assessment of the model depicted in Fig. 4 shows that the GA model provides good solutions, while the GA assisted via a DNN is able to obtain better results by using initial solutions from the DNN rather than pure randomness.

### User Demand Congestion Prediction

#### Congestion in Multibeam Satellite Systems

In satellite networks, the user distributions are unbalanced due to geography and population dispersion. As a result, some satellites have heavy traffic loads, while others face light traffic loads, which often lead to congestion events. We propose a ML-based method that predicts network loads and detects congested areas before they experience congestion. This is especially useful for detecting anomalous behaviors and predicting non-recurrent patterns in a strong nonlinear scenario.

As of today, performance prediction is generally based on trend curves of the previous days. This does not allow either long-term predictions or management of anomalous behaviors. By considering several additional factors, ML techniques can both extend the timeframe in which the analysis is done and identify in quasi-real time irregular traffic patterns that unexpectedly affect the performance.

We investigate user behavior from two different perspectives. The first is the consumed traffic typology (i.e., the used services), while the second refers to the time interval during the day when users mostly consume their traffic (e.g., during a particular day time slot or during the night). We identify 14 service categories (web browsing, social media, streaming protocols, etc.) and 6 time slots (morning, lunchtime, afternoon, dinner, evening, and night).

After scaling and dimensionality reduction using principal component analysis (PCA), we compare representation (t-SNE). As an example, we show a snapshot of this representation in Fig. 5. We tested the hierarchical clustering, k-means, density-based spatial clustering applications with noise (DBSCAN), affinity propagation, spectral clustering, and several clustering algorithms by computing t-distributed stochastic neighbor embedding.

Once the best cluster labeling is computed for each user and for both the service group and the time slot dimensions, we move the focus to network performance. Network performance indicators are considered in a time-series domain, together with other features coming from the clustering phase. These are used to improve predictions with respect to a baseline scenario when no information about the user traffic patterns is provided.

For each identified cluster, we compute the time series of active users. We consider as inactive those users that switch on the modem but do not consume any active traffic during the day, thus being easily identified by an exploratory analysis on the download volume. Active users are instead users that exceed a specific download threshold and consume real traffic volume.

We consider a congested network and address the download speed forecasting. We compare services and the time slots when they are more active. By exploiting relevant trends of users belonging to the same network, we are able to provide additional insights to the specific area of interest. Second, we use ML forecasting techniques to predict network performance indicators, such as average download speed and fill factor index.

### Clustering and Recurrent Neural Network for Congestion Prediction

The objective of this use case is twofold. First, we characterize users by their traffic patterns; we want to cluster users based on their consumed

![](image)

**FIGURE 4.** This figure shows in bars the resulting unmet capacity values in megabits per second. This key performance indicator represents how different the offered capacity is with respect to the requested one. Three different techniques are included; namely, deep learning optimization, genetic algorithm, and genetic algorithm assisted by deep learning. The latter shows the best performance.
three different forecasting algorithms: the recurrent neural network (RNN) with long short-term memory (LSTM), the classic seasonal auto-regressive integrated moving average with exogenous regressors (SARIMAX), and the Prophet algorithm from Facebook. We note that any other network performance metrics can be considered in the analysis since the model structure is highly flexible and easily customizable.

The available time series are split in train and test, with data from January 2019 to October 2019 for training and validation, and on November 2019 for testing. Moreover, we consider the following scenarios:

1. The forecast of the download measured speed with no additional feature
2. The forecast of the download measured speed with network capacity and download volume
3. The forecast of the download measured speed with features derived from the clustering phase

When computing predictions, different parameters are tuned according to the hyperparameters optimization activities within the validation dataset. The identified key performance indicators used to measure the model prediction capacity are percentage (scale-independent) errors, such as mean absolute percentage error (MAPE), in order to deal with different data rates within the network.

As indicated in Fig. 6, we evaluate the download speed prediction when clustered active users time series are considered and compare this scenario with the one characterized by no clustering features. Generally, even though results are comparable, we recognize that LSTM outperforms the other algorithms. Moreover, the improvement of the clustering feature is moderate but evident. We believe that the similarity among the scenarios is due to the scarcity of the time depth and low data quality. We also use the Exponential Smoothing algorithm to compare the above performance with a naive model. We prove that the gain in using any of the ML forecasting models is up to 10 percent on the MAPE indicator with respect to the baseline approach. We leave for future work the investigation of the model performance when a deeper dataset is available as input.

The congestion prediction process is supposed to operate on a regular basis, even daily. Robust data cleaning and preparation should be performed to guarantee adequate data quality and strong data recovery in the case of missing measurements. The algorithms developed for user classification and performance prediction can be deployed on the real network and run on dedicated machines.

CONCLUSIONS
This article deals with the problem of new ML-based data-driven techniques for enhancing current satellite communication operations. We show the potential of these techniques in relevant scenarios employing real-world data. The three procedures (interference detection, flexible payload configuration, and congestion prediction) are initially validated in simulations using real data, and its deployment into the legacy satellite control systems is feasible. All the methods show good performance:

- The interference detector provides a false positive probability performance gain of 44 percent.
- The flexible payload optimization with deep learning is able to decrease the unmet capacity by 32 percent.
- The forecasting method yields a MAPE 10 percent reduction.

Deep learning techniques show great potential in the automatization of the mentioned operations.

ACKNOWLEDGMENT
The authors would like to thank Tomás Navarro (ESA-ECSAT) for his constructive comments.

REFERENCES
[1] R. N. Cronk, P. H. Callahan, and L. Bernstein, “Rule-Based Expert Systems for Network Management and Operations: An Introduction,” IEEE Network, vol. 2, no. 5, Sept./Oct. 1988, pp. 7–21.
[2] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016.
[3] P. V. R. Ferreira et al., “Reinforcement Learning for Satellite Communications: From LEO to Deep Space Operations,” IEEE Commun. Mag., vol. 57, no. 5, May 2019, pp. 70–75.
FIGURE 6. Mean absolute percentage error (MAPE) of the download speed forecast in the three considered scenarios.

[4] P. Henarejos, M. Á. Vázquez, and A. I. Pérez-Neira, “Deep Learning For Experimental Hybrid Terrestrial and Satellite Interference Management,” IEEE 20th Int'l Wksp. Signal Processing Advances in Wireless Commun., Cannes, France, 2019, pp. 1–5.
[5] P. Henarejos and M. Á. Vázquez, “Decoding 5G-NR Communications via Deep Learning,” IEEE Int'l Conf. Acoustics, Speech and Signal Processing, Barcelona, Spain, 2020, pp. 3782–86.
[6] J. L. Rogers, and W. A. Nicewander, “Thirteen Ways to Look at the Correlation Coefficient,” The American Statistician, vol. 42, no. 1, Feb. 1988, pp. 59–66.
[7] D. Whitley, “A Genetic Algorithm Tutorial,” Statistics and Computing, vol. 4, June 1994, pp. 65, 85.
[8] Y. Bengio, A. Lodi, and A. Prouvost, “Machine Learning for Combinatorial Optimization: A Methodological Tour d’Horizon,” arXiv preprint arXiv:1811.06128, 2018.
[9] E. B. Khalil et al., “Learning to Branch in Mixed Integer Programming,” Proc. 20th AAAI Conf. Artificial Intelligence, Phoenix, AZ, pp. 724–31.
[10] H. Sun et al., “Learning to Optimize: Training Deep Neural Networks for Interference Management,” IEEE Trans. Signal Processing, vol. 66, no. 20, Oct. 2018, pp. 5438–53.

BIOGRAPHIES

MIGUEL ÁNGEL VÁZQUEZ [SM’19] (mvazquez@cttc.cat) is a senior researcher at CTTC. He received his Telecommunication Engineering degree, Master’s degree, and Ph.D. degree (cum laude) from Universitat Politécnica de Catalunya (UPC), Barcelona, Spain, in 2009, 2012, and 2014, respectively. He holds a degree in computer science from National Distance Education University. In parallel to his Ph.D. studies, he worked as a research engineer at CTTC from January 2010.

POL HENAREJOS [SM’20] (phenarejos@cttc.cat) is senior research at CTTC. He received the Telecommunication Engineering degree from the Telecommunication Engineering High School of Barcelona at UPC in 2009. In 2012, he obtained his M.Sc. in research on information and communication technologies. In 2017, he obtained his Ph.D. degree (cum laude).

IRENE PAPPALARDO (i.pappalardo@reply.it) is a data scientist at Data Reply IT. She received her B.Sc. and M.Sc. degrees with honors in telecommunication engineering from the University of Padova in 2009 and 2012, respectively. In December 2016, she accomplished the second level master “Big Data in Business” at the University of Rome, Tor Vergata, which allowed her to start a new career as a data scientist.

ELENA GRECHI (egrechi@eutelsat.com) is head of Service Operations at Eutelsat. She obtained her Electronic Engineering degree from the University of Perugia in November 2014 after a brief stay in Brussels, Belgium, for her degree thesis on a directed diffusion routing algorithm for wireless sensor networks. In June 2015, she joined Eutelsat as DVB, IP, and systems engineer, occupying several positions in the company in the Deployment and Operations departments throughout the years.

JOAN FORT ALSINA (joan.fort.alsina@esa.int) is a flight product engineer at ESA. He received his Computer Engineering degree from UPC (IEI FIB) and his Electrical Engineering degree from the French School of Civil Aviation (IENAC ‘00L) in 2003. He also obtained his M.S. in microwaves and optical telecommunications from the Technical National Institute in Toulouse, and later he received an M.S. in space studies from the International Space University in 2006.

JUAN C. GIL MONTORO (jcgil@gmv.com) is marketing product manager for the GMV satellite control suite and GMV Satellite Control and Mission Planning PMO. He received his B.S. and M.S. degrees in chemistry from the Universidad Complutense de Madrid in 1990 and his Ph.D. degree in physical chemistry from the same university in 1997. In 1995 he joined GMV, an international conglomerate of companies where he has devoted the last 25 years to both technical and managerial roles in the area of satellite control.

ROCCO MICHELE LANCELLOTTI (r.lancellotti@reply.it) is a senior data scientist at Data Reply IT. He received his Ph.D. in numerical analysis from Polytechnic of Milan, Italy, at MOX Laboratory, Department of Mathematics. His research involved mathematical modeling and simulation of fluid-structure interaction cardiovascular problems, analysis, fitting, interpretation of experimental and numerical big data, and development of parallel efficient software for high-performance computer architectures.

The congestion prediction process is supposed to operate on a regular basis, even daily. Robust data cleaning and preparation should be performed to guarantee adequate data quality and strong data recovery in the case of missing measurements. The algorithms developed for user classification and performance prediction can be deployed on the real network and run on dedicated machines.