Reinforcement Learning for Satellite Communications: From LEO to Deep Space Operations

Paulo Victor R. Ferreira, Randy Paffenroth, Alexander M. Wyglinski, Timothy M. Hackett, Sven G. Bilén, Richard C. Reinhart, and Dale J. Mortensen

The authors discuss the potential role of machine learning in the link-to-link aspect of the communication systems. An experiment using NASA's Space Communication and Navigation Testbed onboard the International Space Station and the ground station located at NASA John H. Glenn Research Center demonstrates for the first time the benefits and challenges of applying machine learning to space links in the actual flight environment.

Abstract

The National Aeronautics and Space Administration (NASA) is in the midst of defining and developing the future space and ground architecture for the coming decades to return science and exploration discovery data back to investigators on Earth. Optimizing the data return from these missions requires planning, design, standards, and operations coordinated from formulation and development throughout the mission. The use of automation enhanced by cognition and machine learning are potential methods for optimizing data return, reducing costs of operations, and helping manage the complexity of the automated systems. In this article, we discuss the potential role of machine learning in the link-to-link aspect of the communication systems. An experiment using NASA's Space Communication and Navigation Testbed onboard the International Space Station and the ground station located at NASA John H. Glenn Research Center demonstrates for the first time the benefits and challenges of applying machine learning to space links in the actual flight environment. The experiment used machine learning decisions to configure a space link from the ISS-based testbed to the ground station to achieve multiple objectives related to data throughput, bandwidth, and power. Aspects of the specific neural-network-based reinforcement learning algorithm formation and on-orbit testing are discussed.

Introduction

With the U.S. Space Council's recent recommendation and the President's Space Policy Directive 1, NASA is once again setting its sights to return to the Moon and later on to Mars for "an innovative and sustainable program of both robotic and human exploration" [1]. The exciting exploration missions of planetary bodies require careful and comprehensive planning for safe voyage, science rich discovery and learning of the solar system, and inspiration for future generations. One key to these successful missions is the communications between and among robotic and crewed spacecraft, science or relay satellites, surface and space elements, ground stations on Earth, and ultimately principal investigators and mission controllers.

The future of human exploration of space requires new capabilities to support exploration and colonization missions. The future space and ground architecture will provide communications, navigation, and internetworking services for space missions from within Earth's orbit out through Mars and other planetary exploration in deep space. At the center of both the mission satellites and the ground stations, planetary surface elements and relay satellite infrastructure are the communications systems providing seamless operations across the solar system back to controllers on Earth. This multitude of systems is a complex system of systems that relies on compatibility and interoperability among user spacecraft and the communications infrastructure. As these space communication systems become larger and more complex, new methods to control, operate, and enhance the communications infrastructure are needed to more quickly adapt to science opportunities; interpret, understand, and quickly respond to system anomalies; and ease or relieve human operators from constant and detailed interaction with the systems of systems and their multifaceted interaction.

The application of cognitive algorithms to various aspects of the communication systems offers the potential to improve throughput and ultimately data return to Earth from these space missions. Aspects of the communication system that may benefit from improved system intelligence or cognition include link-to-link optimization through adaptive rate changes and configurations based on predictive and learned performance, improved data flow through content-based data routing, data routing to mission control centers from multiple ground or relay connections supported by disruption-tolerant protocols, and system-wide enhancements to scheduling and link configurations based on past performance, predictive performance, scheduling needs and optimizations, and other mission or service objectives.

Software-defined radio (SDR) is one such technology that provides the flexibility and configurability needed for NASA's future cognitive communication systems. SDRs provide the needed on-orbit reconfigurability where a portion of their functions can be updated via software *in situ* while conducting a mission. This reprogrammable capability provides system flexibility and adaptability to change the operating or signaling characteristics, or mitigate or recover from system anomalies.

SDRs have been around for a number of

This work was partially supported by: NASA John H. Glenn Research Center, grant number NNC14AA01A; NASA Space Technology Research Fellowship, grant number NNX15AQ41H; and CAPES Science without Borders scholarship, grant number BEX 18701/12-4.

Digital Object Identifier: 10.1109/MCOM.2019.1800796

Paulo Victor R. Ferreira, Alexander M. Wyglinski, Randy Paffenroth are with Worcester Polytechnic Institute; Timothy M. Hackett and Sven G. Bilén are with The Pennsylvania State University; Richard C. Reinhart and Dale J. Mortensen are with NASA John H. Glenn Research Center.

0163-6804/19/$25.00 © 2019 IEEE

years for ground-based and defense applications. However, NASA's John H. Glenn Research Center (GRC) has been testing the viability of SDRs for space through an experimental communications system called the Space Communications and Navigation (SCaN) Testbed [2, 3]. The SCaN Testbed is a communications research platform installed on the International Space Station (ISS), and it comprises three SDRs with the aim to facilitate on-orbit reconfigurable communication experiments to reduce the risk and increase the experience of using SDRs in space missions. The resultant network topology is illustrated in Fig. 1.

In addition to the flexibility provided by SDRs, the software applications of SDRs should also have flexible, adaptive, and learning properties to enhance performance and operations. Autonomous control systems (i.e., without a human in the loop) for ground station operations and network management exist today; however, moving these autonomous algorithms to the space systems and enhancing them with machine learning or cognitive algorithms will further enhance the capability of the space systems and communications infrastructure. Cognitive algorithms utilize information about the radio channel, network traffic, performance, power, and other sensory information, and feed them into cross-layer intelligence algorithms that implement the radio's cognitive engine. A cognitive engine understands the complex relationship among the system's launched capabilities and changes over time, relay satellite or ground station configurations, real-time link and propagation effects, data content and performance requirements for successful transmission, orbital dynamic effects, and other aspects of the communication system. Ground-based machine learning algorithms are already under development for autonomous driving, consumer behavior, television and targeted advertising, and other applications. Adding machine learning and cognition to space system configurations and operations opens an entire new field of research with early promise to enhance performance, reduce complexity, and ultimately reduce the cost of space operations.

During any space mission, a spacecraft will experience extreme conditions of temperature, radiation, vibration, electromagnetic fields, and resultant gravity forces due to the orbital mechanics. Communication links will experience power supply transients, fluctuation in transmitted power, atmospheric and propagation effects, and ground station variability (pointing and receiver temperature changes). All these factors affect the communication system's performance in terms of link robustness and power consumption. Given that the spacecraft's power generation and storage capabilities are limited, it is important that a cognitive engine manages resources across multiple layers and optimally operates according to the communications demands considering both long- and short-term performance and utilization goals.

NASA GRC teamed with Worcester Polytechnic Institute (WPI) and Penn State University (PSU) to investigate the application of machine learning to functions of the communication system. Applying machine learning to the control of the SCaN Testbed onboard the ISS is one of the first reported applications of machine learning controlling a space system in flight [4–7]. This initial effort, briefly described in the sections below, used a ground-based cognitive engine for real-time learning to decide on changes to the space-based communications link configuration to best meet multiple performance goals and metrics of bit error rate, data throughout, and power through tunable parameters and interacting with the external physical phenomena affecting these performance values.

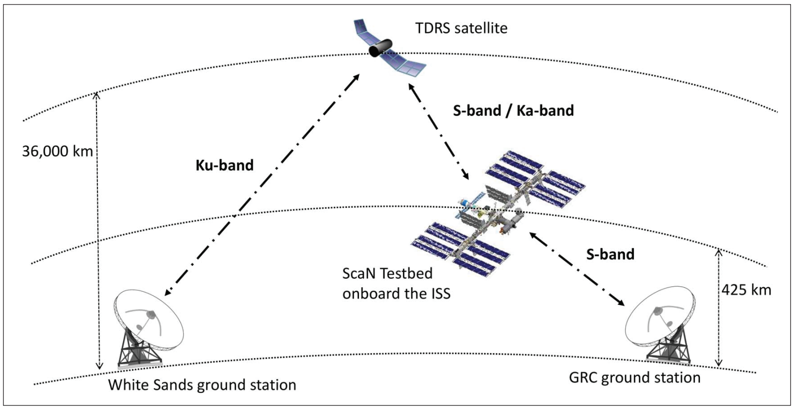

Figure 1. Network topology of NASA's Near-Earth Network links at S-band and Space Network links at S-, Ku-, and Ka-bands between ground stations, the ISS, and satellites at GEO orbit.

The following sections describe the current state of the art in adaptive satellite communications networks, including a brief overview of reinforcement learning (RL) and how it has been applied to satellite communications through multi-objective reinforcement learning (MORL) as a cognitive engine framework. In the conclusions, some future directions for RL-based satellite communications are discussed.

Adaptive Satellite Communications State of the Art

Adaptive coding and modulation (ACM) has been deployed in commercial satellite systems for some time, especially in the broadcast arena with the popular second-generation Digital Video Broadcast for Satellites (DVB-S2) standard [8]. It allows a system to operate more efficiently and mitigate poor channel conditions without wasting excess link margin. NASA has been experimenting with the DVB-S2 standard for link optimization in its space and near Earth networks [9, 10]. These demonstrations have shown the ability to improve user data throughput by several dB compared to NASA's standard fixed modulation and coding waveforms [10, 11]. While usually effective in the near Earth domain, DVB-S2 or other adaptive methods rely on feedback to the transmitter in a timely manner commensurate with channel dynamics, which for deep space links can be a challenge. Consider a link to Mars where the round-trip feedback time varies from 9 to 40 minutes, depending on the relative orbital position with Earth. Cognitive approaches can anticipate changes in the link, making it a fitting approach for such mission scenarios. Anticipating changes can also improve near Earth missions where dynamics of the channel make simple adaptive feedback inadequate. When the adaptation needs to satisfy multiple objectives for the system that vary as the mission progresses, a cognitive approach enables a robust implementation.

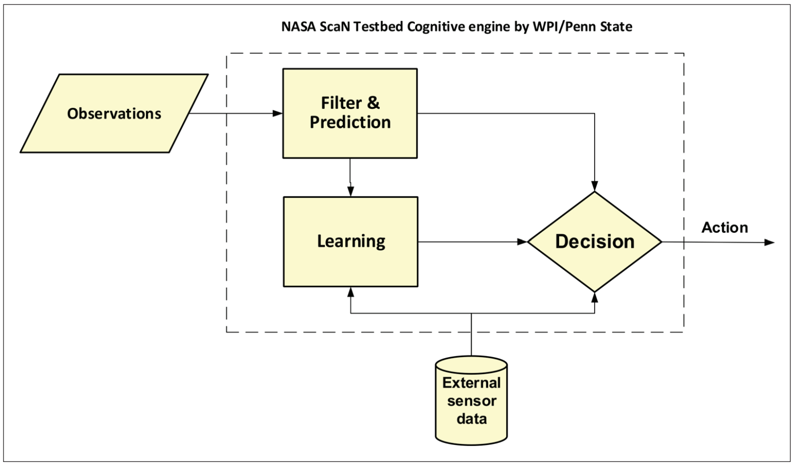

Figure 2. High-level block diagram of the proposed cognitive engine. The inputs are environment measurements from spacecraft sensors and other shared satellite database information. The output is the parameter set configuration for the communication system.

Enabling Cognitive Satellite Communications

Adaptive techniques such as ACM [12, 13], employed by the DVB-S2 standard, are based on lookup tables and show good performance when reacting to changes in the communications channel. However, these techniques do not scale well when multi-sensor information is available (i.e., electronics status, space weather, and other third-party databases), multiple parameters are to be tuned, or multiple long- and/or short-term goals are to be satisfied all at the same time. These characteristics transform the problem into a multi-dimensional one — with multiple inputs and outputs.

Based on these requirements, the satellite communications radio's cognitive engine needs to sense, predict, reason, and adapt to the dynamics of the environment, available resources, and multiple goals. Figure 2 shows a high-level block diagram with the basic cognitive engine components and their interconnections.

At the core of the cognitive engine is a learning capability to enhance operations and allow cross-layer network adaptations to become more efficient. Figure 3 expands the learning algorithm interfaces of the cognitive engine concept, as proposed and validated by WPI/PSU. It is worth noting that the learning block concentrates the majority of the information and builds a knowledge base from which the decision logic collects information to consider when making its decisions.

As an example, sensed information from the spacecraft's electronics, the communications channel, and the available medium access control (MAC) layer parameters could result in achieving a desired performance at different layers. For instance, better packet latency and retransmission control at the network layer, channel access at the MAC layer, and battery power savings indirectly relate to the physical layer resources.

RL is a machine learning technique that is different from supervised and unsupervised learning. Supervised and unsupervised methods learn through batches of examples, either labeled or not, respectively. RL methods, however, continuously learn a task through an agent's interactions with the environment, from which feedback signals are used to measure performance and reinforce the best state transition path between a system's starting and ending positions in an online manner [14]. The RL agent interactions take place via actions that are selected based on received feedback, referred to as rewards. After performing an action, the agent's system is taken into a certain state and receives a reward. Therefore, it is assumed that the problem can be modeled as a Markov decision process (MDP).

By exploring different actions, an agent is capable of discovering different states and thus its performance in a given time-invariant environment (i.e., the functions that maps actions into rewards remains the same). Then, after a certain duration, the agent knows enough states to allow it to exploit certain actions that result in the best rewards known so far.

The challenges related to RL are mainly concerned with defining the agent's behavioral policies (i.e., the policies that guide the agent's learning and action selection over time and mapping between action-state pairs into rewards), assuming information about the reward and new state are made available to the agent after taking an action. In addition to that, a good understanding of the satellite communications requirements, the RF or optical channel, radio capabilities including the tunable parameters, and the available sensor information and measurable parameters are essential to propose and design an RL-based cognitive engine for satellite communications.

On-Orbit WPI/PSU SCaN Testbed Experiment Overview

To the best of the author's knowledge, the flight testing completed during a two-week window from May 2, 2017 through May 12, 2017 is believed to be among the very first fully autonomous cognitive communications experiment conducted using space assets.

The on-orbit experiment had the cognitive engine placed among the ground receiver, while feedback control was sent in the uplink to the SCaN Testbed SDR onboard the space station, where the adaptations were performed by the DVB-S2 transmitter using the same topology shown in Fig. 1.

For each data frame received by the ground modems, the cognitive engine records the previous action tuple with its performance, chooses a new (or the same) action tuple for the next frame, and forwards this decision to the ground transmitter, which is then uplinked to the SCaN Testbed. The space-based software receiver decodes the action tuple and updates the DVB-S2 transmitter in the next frame.

The section below summarizes the proposed cognitive engine framework and provides insights on the design decisions chosen to address the issues mentioned above.

Overview of the MORL Framework for Satellite Communications

The cognitive engine used in this research effort employs the MORL framework previously presented at several conferences and published in [4–7].

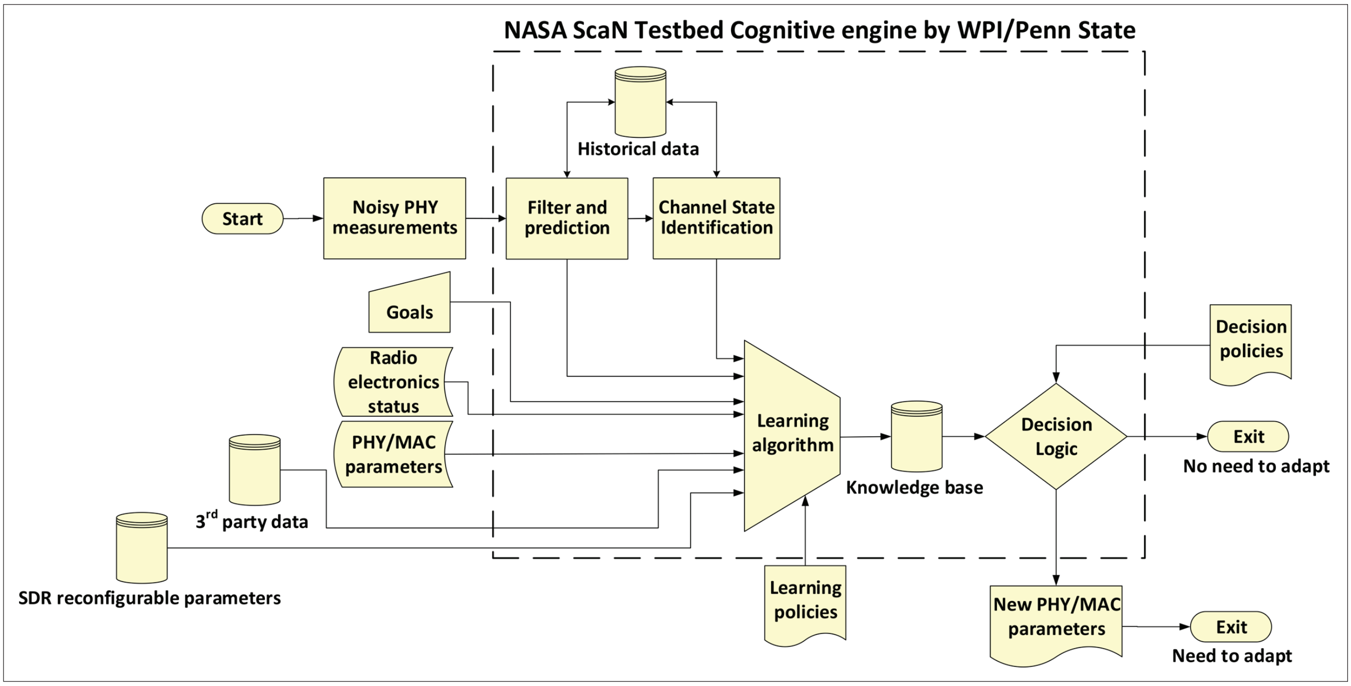

Figure 3. Low-level block diagram of the cognitive engine showing the learning algorithm concentrating the majority of information in order to consolidate knowledge about the environment conditions, communication systems electronics status, and network data to be fed to the decision logic.

The MORL-based framework structure is shown in Fig. 4 and implements the core concept behind the application of the cognitive engine for satellite communication shown in Fig. 3. It extends the principles of classic RL, allowing the cognitive engine to deal with multiple objectives, which may have conflicting goals among themselves.

The Markov decision process comprises states, actions, and rewards. In this work, the states are the communications mission objective parameters: bit error rate (BER), throughput (Thrp), bandwidth (BW), spectral efficiency (Spc_eff), additional consumed power (Pwr_con), and power efficiency (Pwr_eff). The actions are adaptable transmitter parameters: modulation scheme and order, encoding rate, symbol energy and rate, bandwidth, and roll-off factor. The reward is a single scalar given by a weighted sum of the states, in which a set of weight defines a communication mission profile.

Classic RL also relies on past knowledge (e.g., state-action values, decisions, and rewards), usually stored as a Q-table in reference to the Bellman's equation's Q-function that maps state-action pairs into Q-values, based on the policies and reward values being used. In the long term, the Q-table holds decision values that define which action to take in any state. However, this is only valid in a discrete way of defining actions and states while planning an end-to-end path that results in the highest reward.

Satellite communications poses a challenge for classic RL due to the number of dimensions of actions, states, and rewards while operating a dynamically changing environment. Using tables to store the memory of the system, which will quickly grow at exponential rates, becomes impractical when scaling upward. Therefore, the solution known as NN-based RL (RLNN), fully described in [4], leverages the deep learning concept into MORL through both RL exploration and exploitation phases with independent implementations of ensembles of artificial neural networks (NNs). NNs have a chosen fixed number of parameters (i.e., weights and biases) to represent any function. As a result, NNs can scale to represent high-dimensional systems without an increase of memory size, storing only those parameters instead.

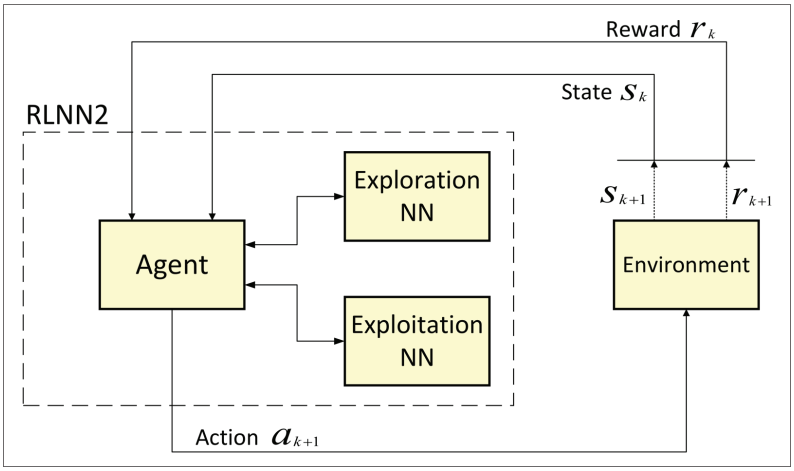

Figure 4. Proposed RLNN block diagram for MORL using deep neural network ensembles. The exploration NN prevents the system from spending time exploring radio parameter combinations that will result in low performance, and the exploitation NN learns actions best suited for the dynamically changing channel conditions.

RLNN Architecture Overview

Inspired by the DQN framework [15], the RLNN autonomously performs online radio resource allocation using a modified Bellman's equation that has actions decoupled from states and only considers immediate rewards.

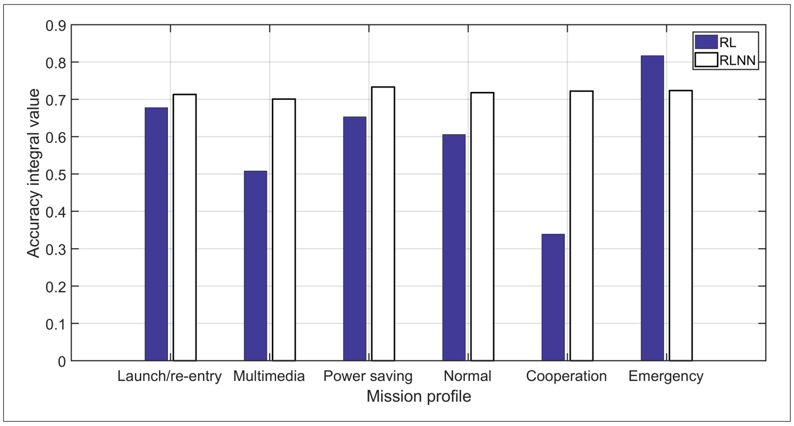

Figure 5. Normalized accuracy comparison between classic RL and the proposed RLNN algorithms for six different mission profiles. Seven different action parameters were autonomously adapted by the algorithms for the same dynamic channel profile. Integral values are computed by the area of accuracy distribution curves. Higher is better. RLNN shows consitent performance across different mission scenarios.

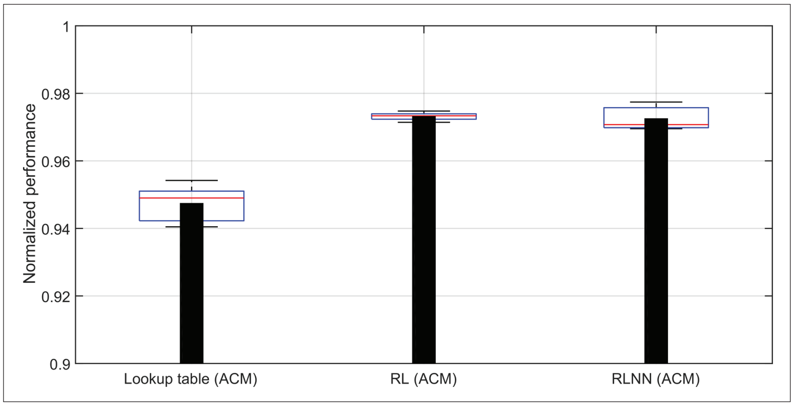

Figure 6. Normalized average performance comparison between the lookup table approach similar to DVB-S2, classic RL, and proposed RLNN algorithms when only modulation and encoding schemes (ACM) are adapted. Boxplots illustrate the minimum, median, and maximum performance levels. Higher is better.

The combination of exploration and exploitation is a key element of RL. Exploration allows the cognitive engine to discover new actions and their corresponding performances in order to gain new knowledge. Exploitation allows the cognitive engine to use the knowledge it has learned to make decisions on which action it should take.

An exponential decay probability defines the exploration turns while alternating with exploitation following a greedy policy. During exploration the agent interacts offline with the exploration NN to classify performance predictions into two groups by a user-defined threshold, and uniformly explores the majority of actions with predicted rewards above the threshold, only occasionally exploring those below it. It is composed of 20 fully connected feedforward NNs with 3 layers each (7-50-1 neurons) trained in parallel by the Levenberg-Marquardt algorithm measured by mean squared error (MSE). The result of this type of guided exploration is that the radio chooses new actions with significantly better performance than if the actions were chosen randomly [7]. This best effort approach prevents the radio from exposing the user to low quality of service (QoS) levels for long periods of time.

During exploitation, the exploitation NN is used as a recommendation engine of action parameters that best achieve the selected mission performance, while the communications channel changes as a combination of orbit mechanics, space, and atmospheric weather. It is composed of a set of parallel NNs, where the set size is equal to the action parameter set. Each unit within the set is composed of 10 fully connected feedforward parallel NNs with 2 layers each (20-1 neurons) and trained using the same methods as the exploration NNs, but each predicts a different action parameter.

As the learning process is independent of the agent operation mode, all performance metrics are collected and stored in a circular buffer alongside excellent action parameter sets to be used during emergency as fallback mode to ensure operational safety. Training by backpropagation is performed periodically and requires intensive computational resource and power. In future embedded deployments, periodicity must be taken into account based on predicted available battery power and computational resources. In this research the training period is defined by the dynamics of the channel, for example, every time there are a certain number of new action and rewards sets in the buffer.

RLNN Simulation Performance Analysis

A comparison of normalized accuracy distribution across six different mission profiles (launch/re-entry, multimedia, DC power saving, normal balanced operation, cooperation with other spacecraft in the same channel, and spacecraft emergency) between the increasing-memory RL solution [7] and the fixed-memory RLNN solution [4] is shown in Fig. 5. RLNN performs just as well, consistently around 0.7, and outperforms RL during the cooperation mission. The accuracy value is the normalized complement of the error between the practical and exhaustive search performances. These results demonstrate the flexibility of the cognitive engine's ability to work with different missions.

In order to compare the proposed RLNN with existing state-of-the-art adaptive algorithms (e.g., the lookup table in DVB-S2 and the classic RL in [7]), we limit both RL and RLNN to optimize for a single objective and to only change modulation and encoding schemes (ACM) on the radio. Figure 6 shows that the normalized performance of RLNN is similar to RL and comparable to the existing lookup table method. An in-depth technical discussion regarding on-orbit experiment results and detailed performance analysis can be found in [6], including software implementation and computational trade-offs discussions.

Conclusions

The potential and scope for machine learning in order to improve satellite communications are large and multi-faceted, from individual links to networks and systems. During normal operations, a space communication system may have different objectives depending on the current conditions of the environment (orbit or flight path), science data, spacecraft health, or interference levels. Mission communications typically strive to maximize data return through high throughput (data rate) and low BER, while minimizing bandwidth and consumed power. Since radio resources onboard a spacecraft are very limited, different mission communication goals might result in competition of available resources in order for an acceptable performance level to be achieved.

The proposed RLNN demonstrated on-orbit with NASA's SCaN Testbed uses a cognitive engine architecture that learns about itself and its environment, enabling it to adapt in real time. It can be applied to many mission types and objectives to autonomously manage radio resources disputed by conflicting communication goals. Being relatively memory-efficient, the implementation is also compatible with limited resource computing platforms onboard spacecrafts.

Future work and application of these technologies will include interference mitigation or other unforeseen environmental or spacecraft changes. These types of capricious impairments may even require autonomous variation of the various parameter weights to maintain robust operations. These types of studies will further increase space communications system robustness by autonomously improving link availability, throughput, and efficiency.

Future work and application of these technologies will include interference mitigation or other unforeseen environmental or spacecraft changes. These types of capricious impairments may even require an autonomous variation of the various parameter weights to maintain robust operations.

References

[1] NASA, "NASA Statement on National Space Council Policy for Future American Leadership in Space," Oct. 2017; https://www.nasa.gov/press-release/nasastatement- on-national-space-council-policy-for-futureamerican- leadership-in, accessed Apr. 18, 2019.

[2] D. T. Chelmins *et al.*, "Lessons Learned in the First Year Operating Software Defined Radios in Space," *32nd AIAA Int'l. Commun. Satellite Systems Conf.*, San Diego, CA, Aug. 2014.

[3] R. Reinhart *et al.*, "NASA's Space Communications and Navigation Test Bed aboard ISS," *IEEE Aerospace and Electronic Systems Mag.*, Apr. 2013.

[4] P. V. R. Ferreira *et al.*, "Multi-Objective Reinforcement Learning for Cognitive Satellite Communications using Deep Neural Network Ensembles," *IEEE JSAC*, Advances in Satellite Communications, vol. 36, May 2018, pp. 1030–41.

[5] T. M. Hackett *et al.*, "Implementation of a Space Communications Cognitive Engine," *IEEE Cognitive Commun. for Aerospace Applications Wksp.*, June 2017.

[6] T. M. Hackett *et al.*, "Implementation and On-orbit Testing Results of a Space Communications Cognitive Engine," *IEEE Trans. Cognitive Commun. and Net.*, vol. 4, Dec. 2018, pp. 825–42.

[7] P. V. R. Ferreira *et al.*, "Multi-Objective Reinforcement Learning–Based Deep Neural Networks for Cognitive Space Communications," *IEEE Cognitive Commun. for Aerospace Applications Wksp.*, June 2017.

[8] DVB-S2 Standard, Nov. 2014; https://www.dvb.org/standards/dvb-s2, accessed April 18, 2019.

[9] J. A. Downey, M. A. Evans, and N. Tollis, "DVBS2 Experiment over NASA's Space Network," NASA Tech. Rep. TM-2017-219524.

[10] J. A. Downey *et al.*, "Adaptive Coding and Modulation Experiment Using NASA's Space Communication and Navigation Testbed," *34th AIAA Int'l. Commun. Satellite Systems Conf., Int'l. Commun. Satellite Systems Conf.*, Oct. 2016.

[11] K. Tasaki *et al.*, "TDRSS K-band Ultra High Data Rate Demonstration," *Proc. AIAA 2008 SpaceOps Conf.*, May 2008.

[12] K. Butchart and R. M. Braun, "An Adaptive Modulation Scheme for Low Earth Orbit Satellites," *Proc. 1998 South African Symp. Commun. and Signal Processing*, Sept. 1998, pp. 43–46.

[13] D. Tarchi, G. E. Corazza, and A. Vanelli-Coralli, "Adaptive Coding and Modulation Techniques for Next Generation Hand-Held Mobile Satellite Communications," *IEEE ICC*, Budapest, Hungary, June 2013, pp. 4504–08.

[14] A. Barto and R. S. Sutton, *Reinforcement Learning: An Introduction*, MIT Press, 1988.

[15] V. Mnih *et al.*, "Human-Level Control Through Deep Reinforcement Learning," *Nature*, Feb. 2015, p. 529–33.

Biographies

PAULO VICTOR R. FERREIRA (prferreira@wpi.edu) received his B.S. and M.S. degrees in electrical engineering from Universidade Federal de Uberlandia, Brazil, in 2010 and 2012, and his Ph.D. degree in electrical and computer engineering from Worcester Polytechnic Institute (WPI), Worcester, Massachusetts, in 2017. He was awarded a Ph.D. scholarship from the Brazilian government program Science without Borders. He has worked as a graduate research assistant at the Wireless Innovation Laboratory, within the Department of Electrical and Computer Engineering at WPI, and as an R&D intern at General Electric Global Research Headquarters. His research interests include satellite communications, machine learning, wireless communications, cognitive radio, and space weather.

RANDY PAFFENROTH (rcpaffenroth@wpi.edu) graduated from Boston University with degrees in both mathematics and computer science, and he was awarded his Ph.D. in applied mathematics from the University of Maryland in June 1999. After attaining his Ph.D., he spent seven years as a staff scientist in applied and computational mathematics at the California Institute of Technology. In 2006, he joined Numerica Corporation, where he held the position of computational scientist and program director. He is currently an associate professor of mathematical sciences and associate professor of computer science at WPI, where his focus is on the WPI Data Science Program. His current technical interests include machine learning, signal processing, large-scale data analytics, compressed sensing, and the interaction between mathematics, computer science, and software engineering, with a focus on applications in cyber-defense.

ALEXANDER M. WYGLINSKI [S'99, M'05, SM'11] (alexw@wpi.edu) is a professor of electrical and computer engineering at WPI, as well as the director of the Wireless Innovation Laboratory at WPI (2007–present) and President of the IEEE Vehicular Technology Society (2018–2019). He received his B.Eng. and Ph.D. degrees in electrical engineering from McGill University, Montreal, Canada, in 1999 and 2005, and his M.Sc. (Eng.) degree in electrical engineering from Queen's University, Kingston, Canada, in 2000. His current research interests are in wireless communications, cognitive radio, machine learning for wireless systems, software defined radio prototyping, connected and autonomous vehicles, and dynamic spectrum sensing, characterization, and access.

TIMOTHY M. HACKETT (tmh5344@psu.edu) is a Ph.D. candidate in the Systems Design Laboratory (SDL), within the School of Electrical Engineering and Computer Science at Pennsylvania State University in University Park. He received his B.S. (student marshal) and M.S. degrees in 2015 and 2017 from Penn State in electrical engineering with an emphasis on communications and signal processing. He was awarded a NASA Space Technology Research Fellowship to fund his M.S. and Ph.D. work. He has worked as an intern at General Electric and the Boeing Company and as a graduate fellow at NASA's Glenn Research Center and NASA's Jet Propulsion Laboratory. His research interests include satellite communications, schedule optimization, and machine learning.

SVEN G. BILÉN [S'90, M'98, SM'08] (sbilen@engr.psu.edu) received his B.S. degree from Penn State in 1991, and M.S.E. and Ph.D. degrees from the University of Michigan in 1993 and 1998, respectively. He is a professor of engineering design, electrical engineering, and aerospace engineering at Penn State, and head of the School of Engineering Design, Technology, and Professional Programs. His research interests include software-defined radio techniques and systems, cognitive radio, and wireless sensor systems.

Richard C. Reinhart (richard.c.reinhart@nasa.gov) is a senior communications engineer with NASA's Glenn Research Center, located in Cleveland, Ohio. He is the principal investigator for NASA's software-defined and cognitive radio flight experiment aboard the International Space Station. He is a principal architect in defining NASA's future communications relay satellite and ground station architecture. He has worked with space communications technology for over 25 years on various satellite, radio, and array antenna technologies. He received his B.S. and M.S. in electrical engineering from the University of Toledo and Cleveland State University, respectively. He has published a number of technical papers and conference presentations associated with the SCaN Testbed, the application of cognitive technologies to NASA, SDR technology, and the Ka-band Advanced Communications Technology Satellite (ACTS). He is a principal author of the SDR Space Telecommunications Radio System (STRS) architecture, now a NASA-wide standard.

DALE J. MORTENSEN (dale.mortensen@nasa.gov) has worked as an electronics engineer at NASA's Glenn Research Center since 1990. He earned his B.S. and M.S. from The Ohio State University and Case Western Reserve University in 1990 and 1995, respectively. The majority of his work at NASA has supported space communications research and flight projects in the areas of free-space laser systems, high-speed digital modems, software-defined radio, and cognitive communication systems. He is a co-author of NASA's Space Telecommunications Radio System (STRS) architecture standard, and now serves as a systems lead for the Cognitive Communications Project.