Using Cognitive Communications to Increase the Operational Value of Collaborative Networks of Satellites

Ryan B. Linnabary
Dept. Elec. & Comp. Eng.
The Ohio State University
Columbus, OH, USA
linnabary.24@osu.edu

Andrew J. O'Brien
Dept. Elec. & Comp. Eng.
The Ohio State University
Columbus, OH, USA
obrien.200@osu.edu

Graeme E. Smith
Dept. Elec. & Comp. Eng.
The Ohio State University
Columbus, OH, USA
smith.8347@osu.edu

Christopher Ball
Dept. Elec. & Comp. Eng.
The Ohio State University
Columbus, OH, USA
ball.51@osu.edu

Joel T. Johnson
Dept. Elec. & Comp. Eng.
The Ohio State University
Columbus, OH, USA
johnson.1374@osu.edu

***Abstract*—Distributed satellite constellations utilizing networks of small satellites will be a key enabler of new observing strategies in the next generation of NASA missions. While it is quickly becoming feasible to establish communication networks among small satellites, a critical question is how these networks can be best utilized to achieve objectives. Small satellite instruments are becoming more capable, but they are still resource constrained (i.e. power, data, scanning systems, etc.); therefore, adaptive instruments that intelligently adjust parameters on the fly are essential for increasing operational value within these constraints. In this context, the purpose of collaborative communication among small satellites is to achieve system-level adaptivity. This dramatically increases the complexity of the control algorithms and decision space in which the small satellites communication networks must operate. We postulate that cognition in the high-level collaborative communication is one approach to both achieve autonomy and to address this complex control space. In this paper, we investigate concepts for how machine learning (ML) algorithms can be utilized in the high-level decision making of a communication system in a distributed satellite mission. This paper will show cognitive communication model with ML and some example case study results.**

***Keywords*—Distributed Satellite Missions, Autonomous Systems, Sensor Network, Sensor Web, OSSE**

I. Introduction

It is envisioned that NASA's future space systems will be composed of large, inhomogeneous networks of small satellites and autonomous platforms [1]. These resource constrained systems, carrying an array of different instruments, will be expected to operate autonomously and collaboratively to achieve mission and science goals. Unfortunately, current and near-future inter-satellite communications are highly constrained in terms of link availability, reliability, power and bandwidth. Although future technologies (such as free space optical links) may alleviate some constraints, it is expected that future instruments will rapidly expand in both data volume and sensor reconfigurability [2]. In this way, it is not sufficient to simply increase the capabilities of the communication links. Rather, it is also necessary to improve the complex decision making that communication systems perform, such as deciding when to transmit, what information is valuable to nodes of the network, and how to adapt local operations following the reception of new information.

Recently, cognitive space communication algorithms have been proposed as a solution to address the complexity of future inter-satellite communication systems [3]. Typically, these cognitive algorithms have tried to address communications at a low level and include decision making regarding modulation, power and bandwidth, and error rate [4], [5]. However, it is reasonable to expect that cognition may also offer an improvement in the complex, higher level decisions of communication in the context of mission and science objectives. At this level, cognition is applied to the operation of the network with the decision making primarily influenced by the constraints of the space communication network links.

In this work, we show results of simulation studies to explore the advantages that cognition could offer for collaborative small-satellite networks. Under a NASA Advanced Information System Technology program, we are currently developing an open-source C++ library for the simulation of autonomous and collaborative networks of adaptive sensors [6]. This library and accompanying utilities allow for the efficient simulation of networks of satellites with realistic constraints in communication, power, and measurements. A key focus of this software is the simulation of sensors that operate adaptively. Adaptive sensors must make intelligent decisions regarding their configuration based on their own measurements as well as the measurements provided by other sensors in a network. However, the extreme complexity of the decision space makes the development of optimal decision-

This research was sponsored by the NASA Advanced Information Systems Technology Program NNH16ZDA001N-AIST.

making systems very difficult. Thus, an approach based on cognition could offer an appealing solution. we investigate how our simulation tools could be useful for production of large training data-sets that capture the operation of collaborative, adaptive networks of small satellites. we then investigate how such a data-set could be combined with machine learning techniques to train neural networks that could make intelligent decisions about when and what to communicate. Results from our investigation will be presented, and the applicability of these methods to future cognitive space communication will be discussed.

II. Collaborative Networks Of Adaptive Sensors

Emerging remote sensor technology indicates a transition from few large monolithic satellites to greater numbers of heterogeneous small satellites. Commercial and government entities have expressed interest in large-scale (1000 or more) small satellite constellations. This transition will involve many system-level changes to how humans interact with and manage these systems. For these complex systems, frequent ground contact (as is currently the case) will be impractical and inefficient. Instead, most decisions related to scheduling or changing parameters will be done autonomously.

Maximized operational value will rely on rapid cognitive adaptation of satellite sensor hardware. This may occur in response to the direct measurements; however, full utilization of adaptive sensors will require collaborative communication among network nodes. Collaborative networking inhabits a complex decision space and is at a low technology readiness level. This research effort has revealed that machine learning and cognition are appropriate tools for exploring the expanded decision space.

III. Cognition and High-Level Communications

A. Cognitive Communications

Many interpretations exist for the term *Cognitive Communications*, so the following definitions provide context for this paper. *Cognition* refers to the act of selecting and carrying out actions based on both specific goals and perception of an external environment. Cognition necessarily includes learning from past experiences and interactions with the environment. Thus, a *cognitive entity* is capable of taking action based on its goals and perception of the environment, potentially learning from the result of its action. It is an intelligent entity that possesses perception, learning, reasoning, and making decisions [7]. *Cognitive Communications* are communications whose operations are in some way dependent on cognition.

Thus far, research into cognitive communication for satellite systems has been predominantly focuses on low-level communication. For example, on-line machine learning was used to optimize the selection of optimal combinations of software-defined radio parameters in [4]. Cognitive digital beamforming is applied to satellite communication in [5]. However, cognition can also be applied to the higher-level aspects of a communication system. For example, cognition could be applied to the intelligent routing of information within an autonomous satellite sensor network [7]. Alternatively, this routing could establish a collaborative improvement in mission science return and achieve, in effect, a system level adaptivity in response to events. Thus, consideration of collaboration to “adaptive remote sensing” and feedback are added to traditional inter-satellite communication.

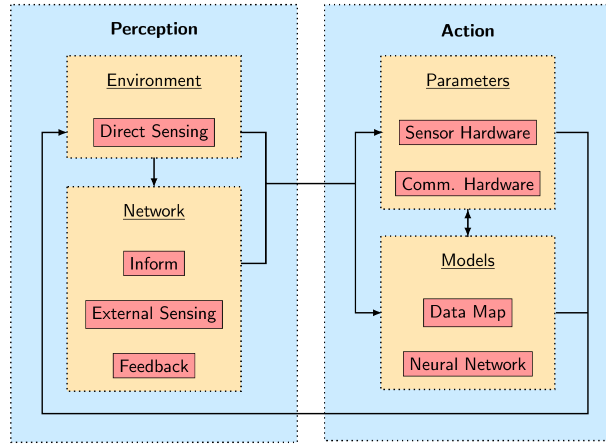


Fig. 1. Cognitive Communications Model

A model for cognitive communications is illustrated in Figure 1, the perception-action cycle for proposed autonomous networks. The model describes a satellite network which communicates collaboratively to perform cognitive routines. Perception is not limited to a satellite’s own measurements; it involves network interaction as well. Nodes in this network take action to learn from perception, improving operations by re-configuring hardware or updating an internal model. In this way, the network relies less on human interaction and becomes *autonomous*.

B. Machine Learning and Cognitive Communications

Machine learning provides computers the ability to learn without being explicitly programmed. Such intelligent machines are capable of accurately predicting and classifying new data. Computers can be programmed to learn autonomously with or without supervision, and often require significant quantities of training data and careful adjustment of model parameters. These techniques are useful for the problems which: require many manual adjustments or long lists of rules; operate in a fluctuating environment; need to process a large amount of data; or have no known optimal solution.

Machine learning techniques are currently being applied in the development of adaptive hardware, resource optimization, and other areas of sensor network design. An opportunity exists to use machine learning to optimize information flow within a collaborative network of satellites. Improvements would increase communications efficiency by increasing the value of data contents and reducing operational resource consumption.

All of the common machine learning algorithms apply to optimization of high-level network management tasks. Regression tasks enable satellites to autonomously adjust parameters for communication, sensing, and on-board data processing. Classification of network nodes based on proximity and capabilities could increase efficiency by informing antenna direction or temporal scheduling. This research involves the application of new and existing machine learning algorithms. Existing algorithms are widely available as third-party Python modules, and will be discussed later in Section IV. Custom algorithms are developed for the C++ library.

IV. Sensor Network Simulations to Support Machine Learning Research

Research in applying machine learning to sensor networks will rely on simulations to validate algorithms both deployed on spacecraft and on the ground. Simulations must provide the means for basic cognitive communication as well as the production of training data for post-processing by neural networks. Training data should include simulation variables which are suspected of correlation to the parameter being optimized. Variables may capture time-series data involving satellite position, health, communication hardware details, sensor hardware details, network connectivity, or other similar parameters.

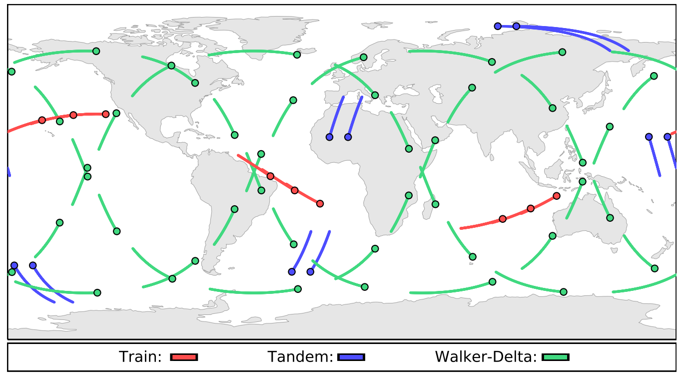
A software tool-set COLLABORATE is under development which is capable of producing the described training data. The tool-set has two main components: first, a C++ development library for observing system simulation experiments; and second, a Python visualization and analysis package for post-processing of data. The project is published to a Git repository under the GPLv3.0 license.

The COLLABORATE library offers a number of unique features valuable to future observing system simulation experiments. At its core, it is a physics engine for satellite position, velocity, and attitude. Power and RF accessories may be attached to satellites and individually oriented. The next level involves rapid constellation design. Standard orbit models described by two-line-element (TLE) sets are provided, copied, and modified to generate novel and interesting constellation patterns. Examples are illustrated in Figure 2(a).

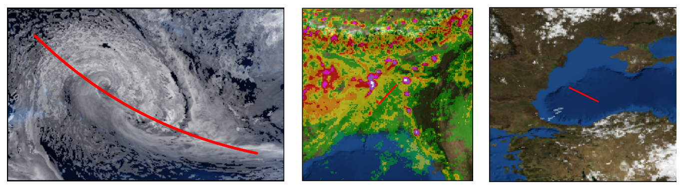
Sensor hardware is attached to satellites as an interface to truth data (NetCDF Nature Run data). This provides a custom modeling environment for real sensor hardware and enables heterogeneous sensor constellations with different capabilities. As a satellite orbits, its pointing vector intersects Earth's surface or an atmospheric layer and samples the underlying data, as shown in Figure 2(b-d).

COLLABORATE is named for its ability to manage collaborative networks of satellites. Its implementation focuses on the high-level communication decision space discussed in Section II. The library employs, in addition to standard C++ components, advanced data structures including trees and graphs to execute predictive route-finding algorithms for efficient communications. For example, line-of-sight wireless channels are captured in a graph, as illustrated in Figure 2(e), the minimum spanning tree.

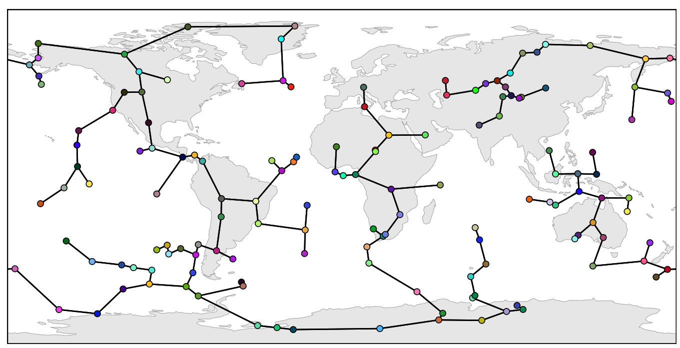
A predictive scheduling algorithm is illustrated in Figure 2(f). Satellite "A" measures cloud depth in the Pacific Ocean (blue line). It then predicts the arrival of a follow-up satellite and relays a message through satellites "B" and "C" to queue "D" for a measurement with a different sensor



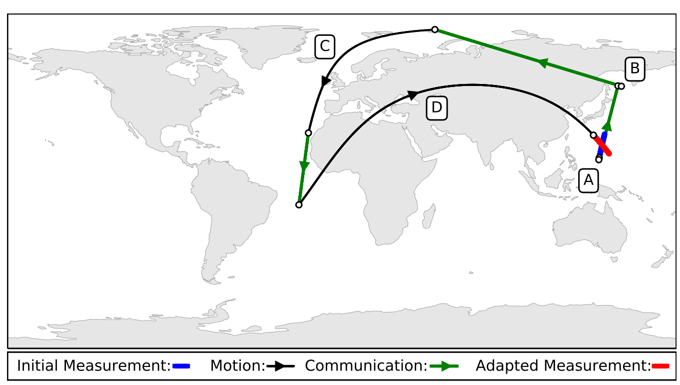
(a) Various constellations defined by orbit patterns



(b) Cloud Depth (c) Precipitation (d) Optical Images

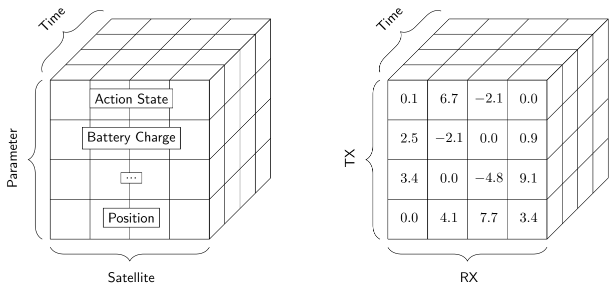


(e) Instantaneous network graph structure (minimum spanning tree)



(f) Collaborative sequence in time

Fig. 2. COLLABORATE software features



(a) Satellite parameter data (b) Network parameter data

Fig. 3. Simulation training data formats

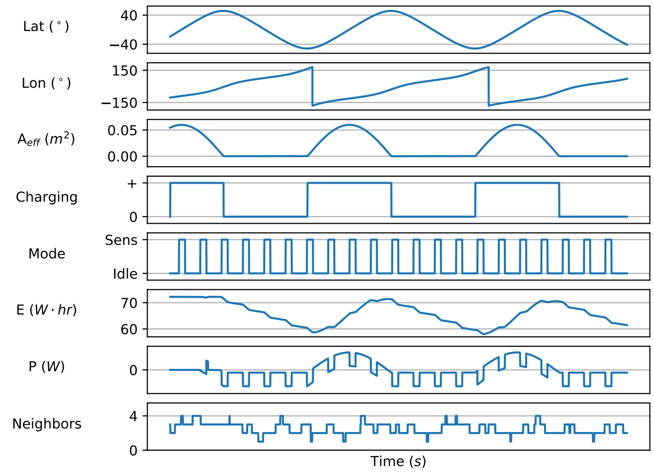


Fig. 4. Visualized simulation data

(red line). Significant work has been done to optimize this algorithm, because it is the main consumer of CPU time and is run regularly in simulation to find routes. It is also the foundation for deployed cognitive communications, as discussed in Section V.

Presently, the included algorithms are iterative and often take minutes to conclude. It may be possible to reduce runtime or optimize routes based on alternative parameters using machine learning. An option is to replace these algorithms with predictive neural networks for advanced regression or classification algorithms. COLLABORATE facilitates this by producing data in standard formats for post-processing.

The software logs simulation data to files accessible by external machine learning tools. COLLABORATE was developed around simple data formats for portability and to promote development of custom analysis tools. Primarily, data is serialized and written to binary files. These formats are well documented and easy to parse in Python or other scripting languages. Included Python packages understand the data formats and can read and store the data for later use as Numpy or Pandas data structures. Examples include time-series data frames or network adjacency matrices (weighted and unweighted). Figure 3 shows several common data structures in memory; one to store satellite parameters; and another for network structures.

Python scripts are provided not only for receiving simulation data at a low level, but also many high level analysis tasks. In fact, all figures in this document were produced using the tools provided by the library. Primary third-party packages used include the following: Numpy, Pandas, Cython, NetCDF4, Matplotlib, Cartopy, Scikitlearn, TensorFlow, and SciPy. These enable post processing for plots and animations or to train machine learning algorithms. Numpy and Pandas provide powerful linear algebra and statistics operations. Cartopy provides extensive map projections and transformations which support visualizing satellite positions and truth data. Machine learning algorithms are available in the Scikitlearn and Tensorflow packages, which interface well with Numpy and Pandas structures.

For example, satellite parameter data (Figure 3(a)) is plotted in Figure 4 to expose and potentially exploit correlations. Several of these seem strongly correlated and many are also periodic. Potential high-level communications optimizations may involve predicting when a satellite has the most visible neighbors (available line-of-sight links). An algorithm for power management scheduling may use the instantaneous charge or power to plan efficient sensor operation.

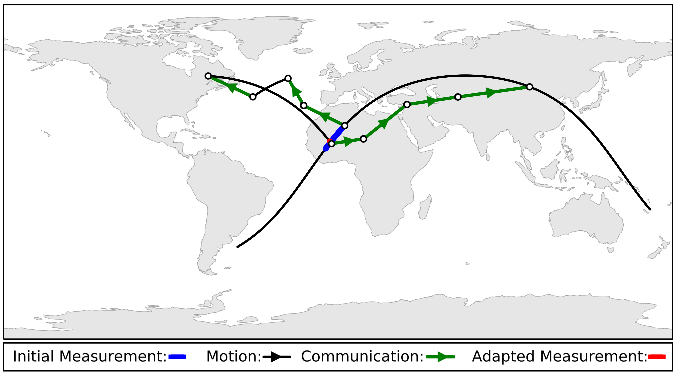
V. Example Case Studies

The following examples demonstrate cognitive communications and machine learning techniques applied to software simulations. First, parametric regression is automated using the COLLABORATE network feedback algorithm. This simulates deployed machine learning in a realistic observing system. Second, satellites are classified based on line-of-sight proximity. This demonstrates the utility of COLLABORATE simulation data for training external machine learning models.

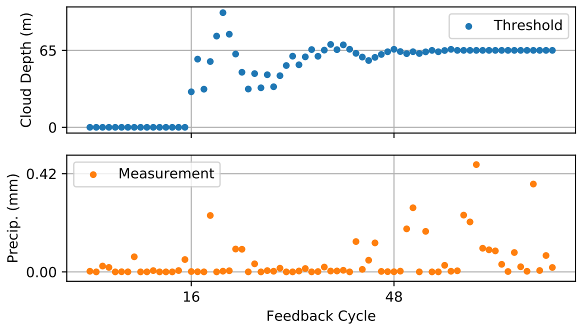
A. *Cognitive Feedback For Autonomous Parameter Regression*

Section IV shows that COLLABORATE provides a network algorithm for predicting optimal routes, such as the one illustrated in Figure 2(f); this can also be used to predict a path for feedback to the original satellite. A single feedback cycle is shown in Figure 5(a). In this plot, a cognitive satellite travels from South America over the Atlantic and senses data over Africa (blue line). Internal processing reveals that the science value of a follow-up measurement exceeds expected resource costs, so the satellite predicts the arrival of another sensor platform. A message is forwarded through the network over the Pacific to queue the next measurement. After a follow-up measurement, data is fed back to the original satellite over the Middle East and China. The contents of this feedback message inform corrective action by the cognitive satellite.

Science value optimization is demonstrated by a 24-hour simulation where this feedback cycle is repeated 75 times. In this simulation, the cognitive satellite performs a regression task to discover the correlation between cloud depth and precipitation. The top plot in Figure 5(b) illustrates regression of the target parameter (cloud depth threshold for non-zero



(a) Feedback route



(b) Autonomous parametric regression for data correlation threshold

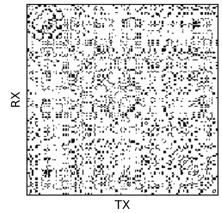
Fig. 5. Cognitive communications feedback

precipitation). Initially the threshold is set to 0 meters and the cloud radar requests follow-up precipitation measurements indiscriminately. For the first 16 regression cycles in the bottom plot of Figure 5(b), 20 percent of the precipitation sensors measured non-zero precipitation. At 16 cycles the satellite begins adjusting the threshold, which converges to a value of 65 meters after 32 cycles. The new threshold improves operational science return by increasing the number of non-zero precipitation measurements to 50 percent.

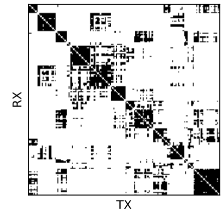
B. Spectral Clustering Using Simulated Network Data

Simulation data described in Figure 3(b) is produced by a simulation and used to train a machine learning (classification) algorithm. The data contains a time series of adjacency matrices with edge weights equal to line-of-sight distances between nodes. A single frame of this data is inverted and normalized to produce an affinity matrix suitable for ScikitLearn's SpectralClustering algorithm. This algorithm identifies normalized cuts in the graph and separates nodes into groups. Figure 6(a-c) show how a graph is sorted and reduced to isolate groups of nodes based on proximity. Figure 6(d) shows the 15 clusters using actual satellite positions.

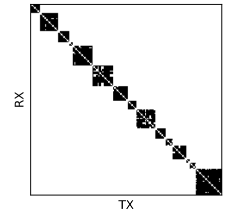
COLLABORATE currently supports one-to-one communication, but future network schemes will require one-to-many links and will likely employ clustering as a single part of its optimization routine. For example a satellite may strategically orient its antenna toward the center of its cluster, maximizing signal strength for its immediate neighbors.



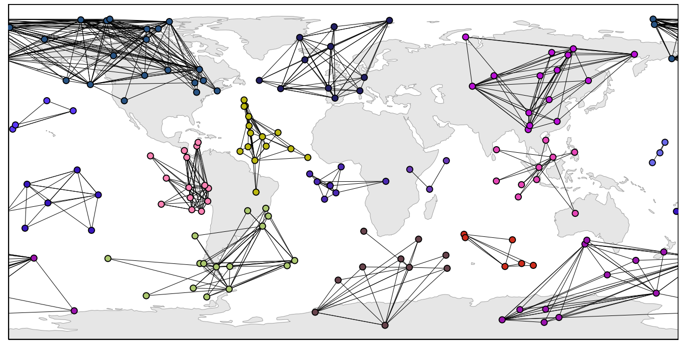
(a) LOS Matrix



(b) Sorted Clusters



(c) Isolated Clusters



(d) Map of isolated clusters

Fig. 6. Spectral clustering by k-means classification

VI. Summary And Next Steps

The COLLABORATE library and utilities facilitate exploration of the expanded decision space for future distributed satellite missions. It includes a space physics engine, constellation design tools, network structures and algorithms, and is capable of producing verbose simulation data for experimental post-processing. In this paper, observing system simulations were augmented by cognitive communication and machine learning techniques. First, a deployed regression algorithm was simulated by the software library. Second, simulation data was logged and used to train an external classification (clustering) model.

Examples presented here indicate the value of cognitive communications and machine learning to future autonomous sensor networks. The regression example demonstrated improvement in science value and optimized mission operations. The clustering example indicates the versatility and portability of COLLABORATE simulation data.

Plans for project development include the following: adapt tools for use on high-performance computers to train neural networks; expand network algorithms for one-to-many wireless links; continue to optimize the route-finding algorithm; and explore additional Python packages for machine learning research and network analysis. COLLABORATE software was written in anticipation of integration with other tools. Currently, significant external work is being done on simulation tools for on-board data processing, resource management, adaptive sensor hardware, and optimized constellations. When combined, these tools will be capable of managing a complex design space for future satellite mission validation.

References

[1] J. Spann, D. Pierce, and A. Petro, "NASA Perspectives on CubeSat

Technology and Highlighted Activities", *SNAP Nanosatellite/CubeSat Subject Matter Expert Exchange*, Brasilia, Brasil, June 2016

[2] A. G. Schmidt, G. Weisz, M. French, T. Flatley and C. Y. Villalpando, "SpaceCubeX: A Framework For Evaluating Hybrid Multi-Core CPU/FPGA/DSP Architectures,", *IEEE Aerospace Conference*, Big Sky, MT, 2017, pp. 1-10.

[3] J. Barnes and W. Eddy, "Machine learning for space communications service management tasks" 2017 Cognitive Communications for Aerospace Applications Workshop (CCAA), Cleveland, OH, 2017, pp. 1-4.

[4] P. V. R. Ferreira et al., "Multi-objective reinforcement learning-based deep neural networks for cognitive space communications" 2017 Cognitive Communications for Aerospace Applications Workshop (CCAA), Cleveland, OH, 2017, pp. 1-8.

[5] Wenhao Xiong, J. Lu, X. Tian, G. Chen, K. Pham and E. Blasch, "Cognitive radio testbed for Digital Beamforming of satellite communication" 2017 Cognitive Communications for Aerospace Applications Workshop (CCAA), Cleveland, OH, 2017, pp. 1-5.

[6] R. B. Linnabary, A. O'Brien, G. E. Smith, C. D. Ball, and J. T. Johnson, "Open Source Software For Simulating Collaborative Networks Of Autonomous Adaptive Sensors", submitted to *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, January 2019.

[7] G. E. Smith, A. E. Mitchell, C. D. Ball, A. O'Brien, and J. T. Johnson, "Fully adaptive remote sensing observing system simulation experiments", in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium* , July 2018, pp. 5839-5842.