International Journal of Wireless Information Networks (2022) 29:257–268
https://doi.org/10.1007/s10776-022-00572-9

Sharing Wireless Spectrum in the Forest Ecosystems Using Artificial Intelligence and Machine Learning

Sonia Naderi[1,2,3] · **Kenneth Bundy**[1,2,3] · **Thayer Whitney**[1,2,3] · **Ali Abedi**[1,2,3] · **Aaron Weiskittel**[2] · **Alexandra Contosta**[3]

Received: 18 June 2021 / Revised: 29 April 2022 / Accepted: 15 July 2022 / Published online: 9 August 2022

Abstract
Intelligent management of power and spectrum is the most important ingredient in creating wireless sensor networks with high reliability and longevity. The main application under study in this paper is accurate monitoring of forest ecosystems using high spatio-temporal resolution. High cost of the current systems and their power consumption limits wide spread use of these systems limiting the accuracy of current models. This project utilizes artificial intelligence and machine learning to learn the changes in the wireless network and environment, producing power efficient systems that are low cost to enable large scale monitoring. The proposed system was built at the University of Maine's Wireless Sensor Networks (WiSe-Net) laboratory in collaboration with University of New Hampshire and University of Vermont researchers for soil moisture measurement with provision to include other sensor types at later stages.

Keywords Artificial Intelligence · Wireless sensors networks · Forest models · Soil moisture

1 Introduction

Measuring forest ecosystem properties and processes has become increasingly complex, involving a variety of data collection systems, software, and computing environments. Intelligent management of power and spectrum is the most important ingredient in wireless communications and creating wireless sensor networks [1, 2]. Sensor nodes, or small affordable devices with limited computational power and memory [1], may enable high-resolution forest ecosystem monitoring if they are integrated into a network that minimizes power consumption. Artificial Intelligence (AI) and Machine Learning (ML), in particular Reinforcement Learning (RL), can provide the requisite tools for this network integration. Artificial intelligence provides the background for information systems with a focus on increased systems automation and better systems control. Using Artificial intelligence for environmental monitoring has drawn significant attention in recent years [3–7]. In Wireless Sensor Networks (WSN), ML techniques can be used to avoid the need for re-programming [8]. This technique is useful in WSN for deploying the nodes in extreme environments and collecting the data from unreachable and dangerous locations. In addition, WSN extract a large amount of data which may not be properly correlated, and ML techniques can be used to extract data from different levels of abstraction.

Here we propose a WSN to monitor soil moisture, which has been increasingly recognized as an important ecosystem property in forested and agricultural systems alike [9, 10, 11, 12, 13, 14], inspiring the establishment of both soil moisture monitoring networks [15–18] and large, freely available soil moisture databases [18]. Despite the importance of measuring soil moisture and its distribution across the landscape [19], the cost of commercial soil moisture sensors remains prohibitive. We have designed a low-cost system that includes wireless sensor nodes managed by an AI engine for power efficiency. Although we focus here on soil moisture measurement, the same methodology could be extended to other types of sensors with proper power and frequency optimization.

The rest of this paper is organized as follows. The overall system block diagram is presented in Sect. 2. ML for wireless sensing is expressed in Sect. 3. Experimental results

✉ Ali Abedi
ali.abedi@maine.edu

1 Electrical and Computer Engineering Department, University of Maine, Orono, ME, USA

2 Center for Research on Sustainable Forest, University of Maine, Orono, ME, USA

3 University of New Hampshire, Durham, NH, USA

are presented in Sect. 4. Concluding remarks are presented in Sect. 5.

2 System Design

A WSN is a distributed sensor network to monitor physical or environmental conditions, such as air temperature, relative humidity, soil temperature, and soil moisture to cooperatively pass data through the network to a centralized processing location or act on the information in a distributed manner [20]. Wireless soil moisture sensor network refers to WSN with the networking of soil moisture sensors. These networks are bidirectional and also allow control of sensor sampling rate and transmit/sleep state. Figure 1 shows the system block diagram of the proposed soil moisture sensing system. Each block is explained in the following subsections.

2.1 Soil Moisture Sensor

One important factor affecting the growth rate of forests is the available moisture in the soil [13]. In addition to the availability of water for the plants themselves, the level water in the soil affects the usage of nitrogen uptake by the roots and the oxygen level at the roots [21]. There are two types of soil moisture sensors, contact-based and contact-free.

In contact-based method, the detection area of the sensor needs to be touched directly with the detection media, i.e., the soil . Contact-based sensors have various methods based on detection parameters such as capacitive sensors [22], heat pulse sensors, and fiber optic sensors [23]. With contact-free sensors, there is no need to contact the detection media that is being detected. Contact-free sensors include passive microwave radiometers, synthetic aperture radars, and thermal methods [24, 25]. Contact-free sensors are more expensive and more complicated compared to contact-based sensors.

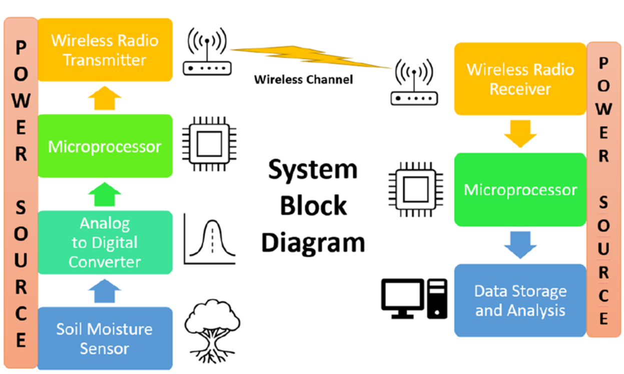


Fig. 1 System block diagram with the soil moisture sensor as a typical example

The standard way to determine soil moisture is the thermogravimetric method which is introduced in [26]. In this method, the weight loss of soil is measured after oven drying of soil with known mass at 105 °C. The main issues with this method are that they are very time consuming and they can not be repeated because they are destructive measurements.

Over the past several decades, these destructive methods have been replaced by electronic devices such as capacitance, impedance, dielectric and time domain reflectrometry sensors [27]. Different soil moisture measurement techniques have been proposed in the literature [28, 29, 30]. For instance in [30], the authors proposed a way for measuring soil moisture content by monitoring electromagnetic radiation of soil, which depends on sensitivity of microwaves to soil moisture. Impedance soil moisture sensing technology involves inserting separate rods into the soil and changing conductivity by altering water content [31]. This method is based on changing the soil conductivity by changing the water content of the soil. Frequency domain sensors has been proposed in [32]. These kinds of soil moisture sensors measure soil impedance changes because of the water content variations. These sensors are available as single and multi sensor probes which offer different measuring techniques [33, 34]. Other methods include fiber optic sensors [35, 36], dye doped plastic fibers [37], ceramic sensors [38], and neutron scattering method [39].

In this project, we are using the DFRobot SKU:SEN0193 which measures soil moisture levels by capacitive sensing rather than resistive sensing, which is more durable, stable, and most importantly low power. It is made of corrosion resistant material and includes an on-board voltage regulator with an operating voltage range of 3.3–5.5 V enabling easy connection to a low voltage microprocessor with support for both 3.3 V and 5 V. *This was selected over the Adafruit STEMMA I2C Capacitive Moisture Sensor capacitive soil moisture sensor and the Grove Capacitive Soil Moisture Sensor based on the criteria outlined above.*

2.2 Analog to Digital Converter (ADC)

Since the selected sensors are analogue devices, it is necessary to convert the sensor output to digital format, readable by the microprocessor that can only accept digital inputs.

To gather sufficient information on the temporal and spatial variations and characteristics of soil moisture, it is highly desirable to take measurements at a sufficiently high frequency as determined by ecological research questions. A competing objective is to have the network function in an automated fashion for as long as possible, since such networks are typically deployed in forests without immediate access to power or human intervention. This requires us to reduce the active operating times of the wireless nodes to conserve energy, even when renewable power sources are used. We need to make judicious decisions in measurement scheduling, i.e., when is the best time to take a measurement, so as to minimize the total amount of time the node needs to be active in actuating the moisture probes and in data transmission, while still satisfying the monitoring objective, i.e., achieving a desired level of accuracy (as determined by the ecological models) in the estimated soil moisture evolution using the measurement data collected.

The output values of soil moisture sensor varies from 0 to 100 representing the lowest and highest soil moisture, respectively. The Texas Instruments Launchpads has 12-bit Analog to digital converter (ADC) and its sampling rate is 200 ksamples/s. It means the resolution or the number of intervals of this ADC is equal to 4096 and the dynamic range is 72dB. The least significant bit (LSB) can be calculated as full scale range of the sensor output voltage divided by number of intervals which is 4096. Since the sensor output values vary between 0 and 100, the LSB is equal to 0.024 and the quantization error in our ADC is around 0.012.

2.3 Microprocessor

The computational logic is responsible to handle on-board data processing and manipulation, temporary storage and data encryption. The faster and more powerful processors usually have a higher energy consumption and cost. Processors with high code density and different operational modes like active, idle, nap and sleep modes to preserve energy are required.

There are different microprocessor options such as Intel 8051, Microchip PIC, Atmel AVR and TI ARM. Among these microprocessor options, ARM processors are widely used in consumer electronic devices. Because of their reduced instruction set, they need fewer transistors, which enable a smaller die size of the integrated circuitry (IC). The ARM processors' smaller size and lower power requirements makes them suitable for increasingly miniaturized devices.

In our project, we are using Texas Instruments CC1310 device which is a wireless microcontroller unit (MCU) with an ARM Cortex-M3 microprocessor. The ARM Cortex-M3 processor is a 32-bit processor for low-cost high performance applications. The ARM Cortex-M3 processor family was selected because they are optimized for cost and are energy-efficient. These processors have been used in a variety of applications, including a variety of edge devices, industrial control, and everyday consumer devices. The processor family is based on the M-Profile Architecture that provides low-latency and high reliability in embedded systems. The Cortex-M3 processor provides a high-performance, low-cost platform that meets the system requirements for low-power consumption and high reliability.

2.4 Radio Module

Radio modules are required to enable sensor nodes to communicate with each other and to the base station. We are using a Sub 1 GHz radio module which provides a reliable transceiver with one built-in antenna at a reasonable cost. Sub 1 GHz RF operates in the ISM spectrum bands below 1 GHz—typically in the 769–935 MHz , 315 MHz and the 468 MHz frequency range. They offer more range than the 2.4 GHz. Sub 1 GHz wireless transmission offers 1.5-2 times more distance coverage than the 2.4 GHz spectrum. Also, the Sub 1 GHz wireless spectrum has a long range mode that is well suited to this application. Wireless Sub 1 GHz RF needs a lower power signal from the transceiver compared to the 2.4 GHz spectrum to get the same output power signal at the receiver.

2.5 Antenna

Five antenna types were considered. Each antenna was subjected to the same range testing. However Antenna 1, a CR2032 PCB Antenna, had such a poor overall performance, such that it was irrelevant to include in this report (Tables 1 and 2).

Especially in regards to power efficiency, this provides close, but not exact expectations of the system. Antenna 3, a compact PCB helical antenna, and antenna 4, an orthogonal array of two helical antennas, performed similarly in range testing. Antenna 5 was the worst performing, with a range of under 100 feet before falling below a level that was unreadable. Using a Received Signal Strength Indicator (RSSI) cut off value of −75 dBm, the board antenna achieved a working distance of 250 ft, and for now, we will use the 250 ft. result to design our network grid. Figure 12 shows RSSI of each antenna at each distance measured.

Future designs will implement a compact PCB helical antenna, demonstrated with antenna 3, as it allows for the possibility of increasing the current 250 ft. range and reducing the overall size of the device.

Table 1 Antenna Metrics [40]

Antenna option	Directivity (dBi)	Effective radiated power (%)
2	3.92	46.61
3	4.13	63.05
4	4.39	31.33
5	4.16	46.83
On-board antenna	4.47	80.38

2.6 Software Design

The software for the system was created using the Energia framework for Texas Instruments microcontrollers. This was selected to allow for the system to be easily extensible to a variety of sensors and enable the implementation of machine learning models on the nodes themselves via TensorFlow lite. The system control flow is detailed in Fig. 2. This general system control allows for maximum power conservation by only transmitting data that is significantly different than what was previously transmitted. Furthermore, the system only transmits when there is sufficient power. The ML inference is also detailed in the diagram; this step constitutes the routing and transmission decisions outlined in the next section.

3 ML for Wireless Sensing

3.1 Introduction

We test three main approaches for controlling the network. First, a baseline model without any AI is developed. Second, a single AI is developed to control the entire network. Third, a multi-agent approach is adopted.

In the initial baseline phase, a battery-agnostic round-robin approach to network traffic scheduling is used for scheduling network traffic. This is done in a battery-independent manner, allocating network resources to the nodes in equal proportion, without regards to their battery level or the size of their transmission queue.

When using the multi-agent approach, the nodes must cooperate to share channel resources and power. This means that the control problem falls in the purview of multi-agent RL—the nodes must make individual decisions about a shared environment [41].

However, the states of the other nodes are not directly observable by any node. This means that this is a Partially Observable Markov Decision Process (POMDP) [42]. In order to control the sensor network, an innovative AI algorithm was developed using RL. This system placed each autonomous sensor node under the control of a ML system which controlled the transmission scheduling for the node.

Multi-agent RL for POMDPs is one of the least studied and most difficult areas of RL [42]. A number of approaches have been proposed for this type of problem [41].

A great deal of success has been achieved in this area using game theoretic approaches [43, 44]. However, the application of these approaches can be challenging, and so we propose using ML to allocate these resources.

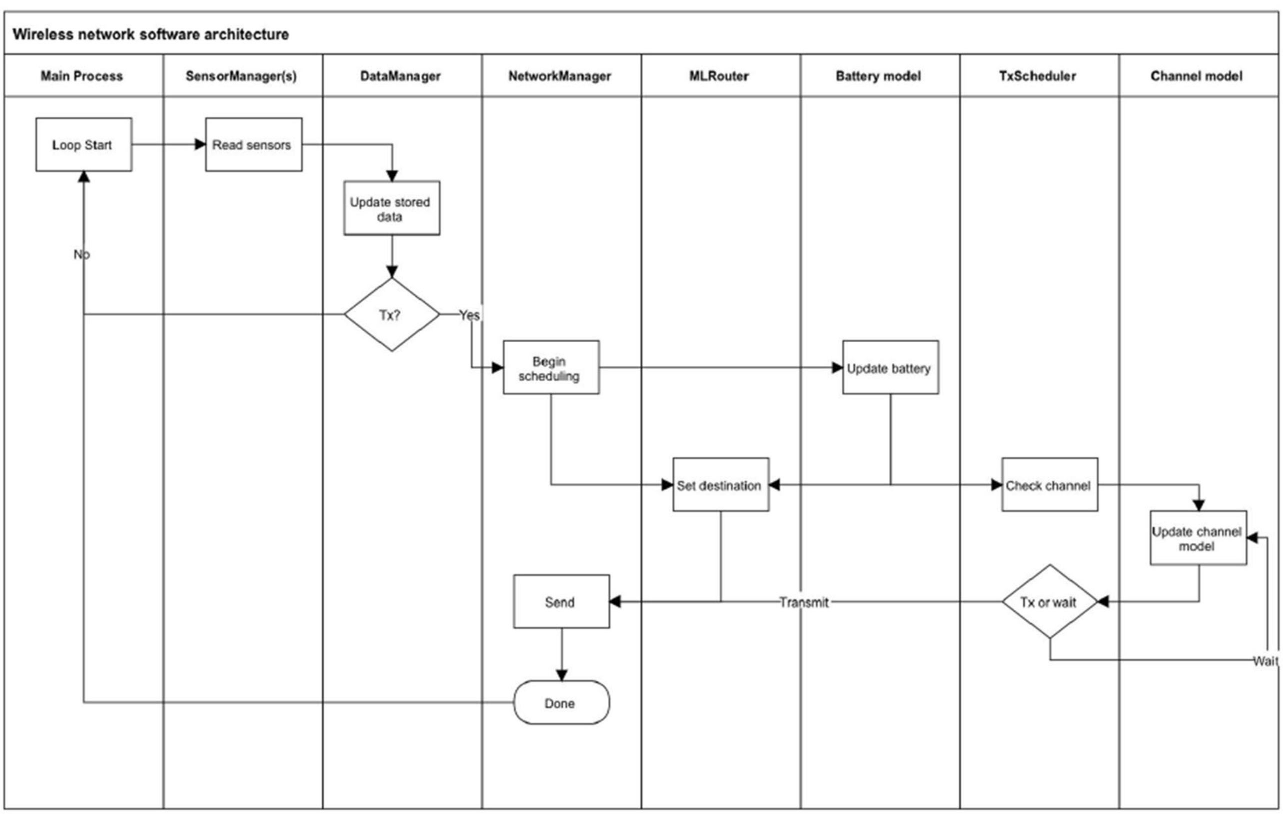


Fig. 2 Software control flow for the proposed system

Specifically, we use a RL as it is the branch of ML concerned with game theoretic tasks and system control [45].

3.2 Methods

3.2.1 Environment

We began by creating an environment for the agents. This included star network and relay network configurations, Quadrature Phase-Shift Key (QPSK) modulation, and fading channel simulation via [46]. This source code is available on GitHub. The network was designed using the OpenAI Gym API.

The environment simulated a star network architecture, with several nodes transmitting to a central base station. Depending on the simulation, different observations were returned to the agent for decision making. These inputs are discussed in each section.

Once the simulation was created, the following methods were implemented using the PyTorch ML Framework:

- Baseline Model
- Q-learning
- Deep Q-Network (DQN)
- Multi-agent Deep Q Network

One primary motivator for using Q-value based approaches is that it allows for caching the outputs of the system, and may avoid deploying the entire neural network in the field. Policy-based approaches are also being developed in future work (for example, A2C as used in [42]).

3.2.2 Baseline Model

The baseline model used for the star network was a round-robin architecture. The process was as follows: the center node requests all data from each node in sequence. This means that node one transmits one packet to the base station, then node two transmits all its data, and so on for each node in the network.

Algorithm 1 Baseline Network Control

```
for node in network do
    send request to node
    if request Received by node then
        send packet to base station
    end if
end for
```

3.2.3 Deep Q Network

The reward function used in training the deep Q network is as follows:

$$R(a,s) = \begin{cases} -1 & \text{battery } \leq 0 \\ 0 & \text{no transmission} \\ 1 - \frac{n_{node}}{N_{total}} & \text{otherwise} \end{cases} \tag{1}$$

where n_{node} is the number of times this has transmitted, and N_{total} is the total number of transmissions made across the network as a whole. This reward function was developed to ensure that all nodes should transmit. Preliminary experiments showed that the DQN would tend to neglect packets from all but one or two nodes. Changing to this reward function rectified the problem (Fig. 3).

The observation by the agent was the battery level and the number of transmissions at each node. Although the system was able to coordinate taking turns from this input, we encountered an implementation problem as the information is not available to individual sensor nodes. This could potentially be address using a Q-value or a multi-agent RL approach using only local information.

The network architecture selected was simple: seven fully connected layers with exponential linear unit (ELU) activation functions [47]. Architectures with Rectified Linear Units (ReLU) activation functions often failed to converge. Changing to from ReLU to ELU activation functions caused the models to converge while holding other hyperparameters constant.

The convergence dynamics of the DQN are shown in Figs. 4 and 5. The DQN is a well studied architecture, and several enhancements could be made including the use of Huber Loss in place of the mean squared error (L^2) loss, the use of of a target network, etc.

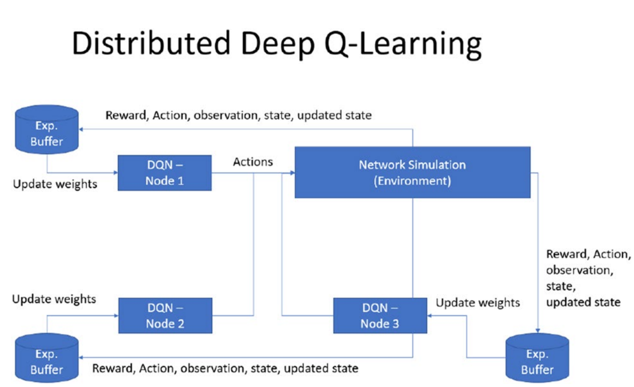


Fig. 3 Distributed deep Q-learning block diagram

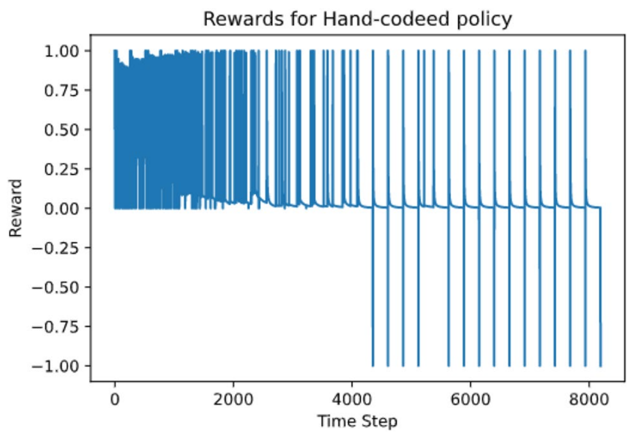


Fig. 4 Convergence of deep Q network implementation

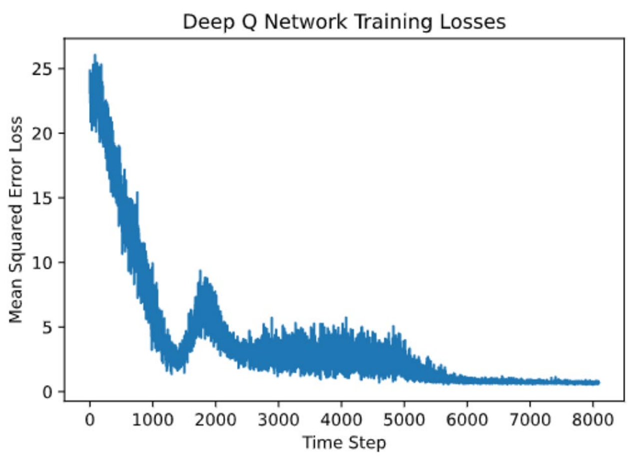


Fig. 5 Convergence of deep Q network implementation

3.2.4 Multi-agent DQN

For the multi-agent system, each agent (node) observed the environment based on the received waveform at the antenna.

The neural network for this approach would have to be run on sensor nodes with limited computing resources, and would have to be run every transmission. This creates a strong incentive for the network architecture to be as small as possible. We were able to get good performance with as little as four layers by using two one-dimensional convolutional layers followed by two fully connected layers. However, much better results were obtained with a deeper architecture. This can be seen in Fig. 6. All layers used ELU activation functions.

The multi-agent system required a modified reward function. The same reward was given to all the nodes, but packet collisions had to be addressed.

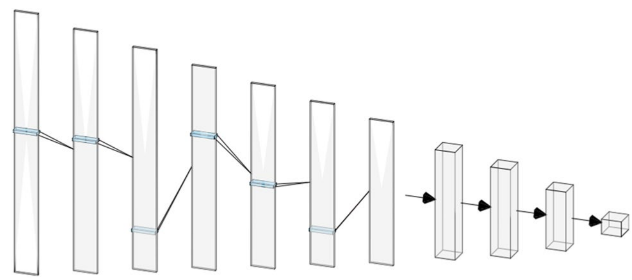


Fig. 6 General architecture of small single agent

Table 2 Multi-agent neural network architecture (N is batch size)

Layer	Description	Tensor Dimensions
0	Input	N, 1,10000
1	1-D Convolution	N, 1,4999
2	1-D Max Pool	N, 1,2501
3	Dropout (p=0.2)	N, 1,2501
4	1-D Convolution	N, 1,1250
5	1-D Max Pool	N, 1,626
6	Dropout (p=0.2)	N, 1,626
7	1-D Convolution	N, 1,312
8	1-D Max Pool	N, 1,157
9	Dropout (p=0.2)	N, 1,157
10	Linear/fully connected	N, 64
11	Linear/fully connected	N, 16
12	Linear/fully connected	N, 2

$$R(a,s) = \begin{cases} -1 & \text{battery } = 0, \text{collision} \\ & \text{or no transmission} \\ 1 - \frac{n_{node}}{N_{total}} & \text{otherwise} \end{cases} \tag{2}$$

This reward function differs from the previous one in a few key ways. First, it penalizes packet collisions, which could not happen in the previous architecture. Further, it also penalizes all the nodes if none of them transmit. Without this penalty, our research suggested that all the nodes would choose not to transmit to avoid packet collisions. This was clearly not the desired behavior, so the reward function was modified to address this problem.

The input to the agents was the simulated waveform at the receiver and the nodes battery level. This included the transmitted waveform from each other node, with multi-path fading, distance-based attenuation, and Gaussian noise. These modulated waveforms allowed the nodes to 'overhear' other nodes' communications to determine when the channel is free.

Reward and loss from the training of the system are shown in Figs. 7 and 8, respectively. The system was trained using an epsilon-greedy method.

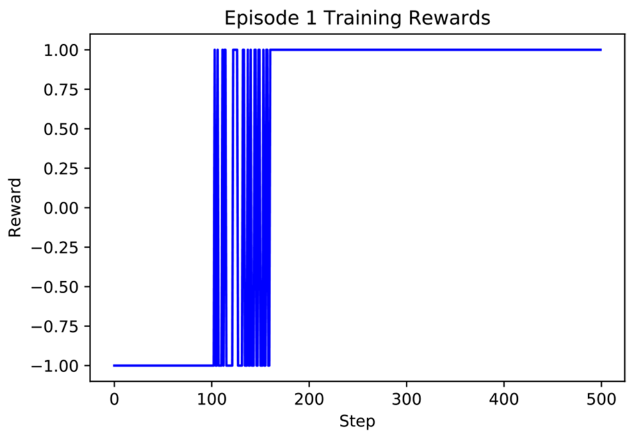


Fig. 7 Episode 1 training rewards for multi-agent DQN

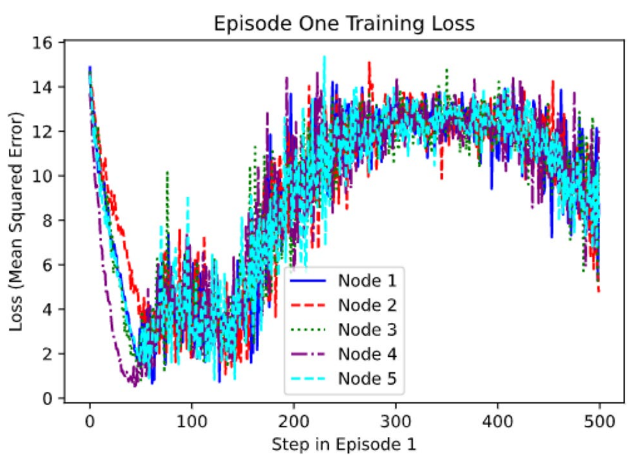


Fig. 8 Episode 1 losses for multi-agent DQN

Multi-agent systems come with difficulties with convergence and are difficult to optimize [41]. Initial attempts to create a multi-agent system converged, but the majority of nodes would not speak, and the last node would transmit as often as possible. This strategy lead us to penalize this in the reward function, but it is worth noting that this was detrimental to the convergence dynamics of the system; failures for the system to converge occurred during training, but maintaining a higher ϵ (more exploration of the state space) improved the convergence dynamics.

Further, another method allowed the system to converge. When choosing actions randomly, each of the k nodes chose to transmit with probability $p = \frac{1}{k}$ and chose not to transmit with probability $1 - p$. This meant that on average, one node is transmitting to the base station at any time step, when actions are chose randomly. When the nodes transmitted and held their transmission with equal probability (that is, $P(transmit) = P(nottransmit) = 0.5$) the learning dynamics either did not converge, or converged to the situation where no nodes transmitted. The altered probability of transmission, we hypothesize, lead to better exploration and allow the system to learn the desired behavior by allowing the system to find the desired behavior at random.

3.3 Results

3.3.1 Power Use

The primary goal of the system was to optimize power consumption of the network system. To this end, we compared the power usage across the three architectures; i.e. baseline model with no AI, single AI model, and multi-agent model.

There are a number of other considerations for choosing the best approach. The multi-agent DQN requires each node to run it own deep neural network, and perform one forward pass at each time step. This may not be practical for all applications, as this requires significant computational resources.

The centralized control approaches exploit the star network architecture to place the base station in control of the entire network. This means that only the base station would need to complete the forward pass of the neural network, allowing the nodes to conserve power.

A comparison of the power used by the three approaches is shown in Fig. 9. The baseline model, as it was battery agnostic, lead to nodes often exhausting their batteries.

3.3.2 Network Resource Use

As you can see in Fig. 10, the baseline network called on the nodes to transmit far more often. This means that the the DQN and multi-agent DQN are much better at maintaining high battery levels than the baseline that did not include awareness of the battery.

However, the best method for optimizing the received reward was the Multi-Agent DQN as shown in Fig. 11. This

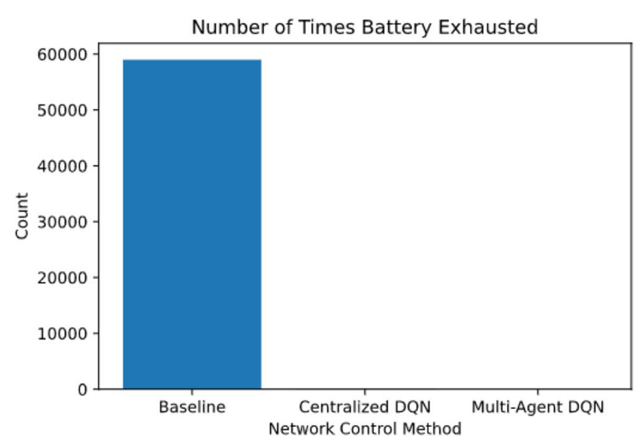


Fig. 9 Comparison of power usage by all three methods

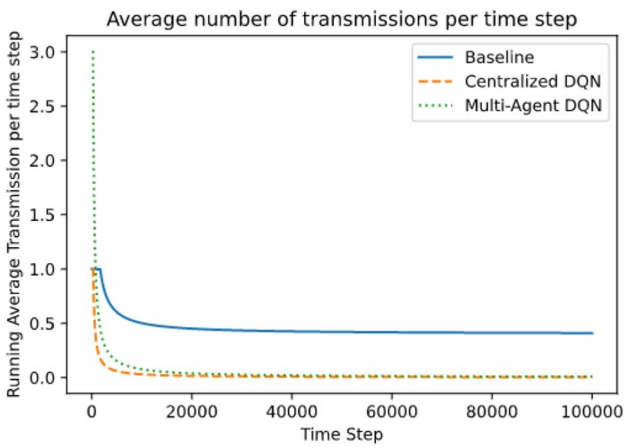


Fig. 10 Comparison of network utilization

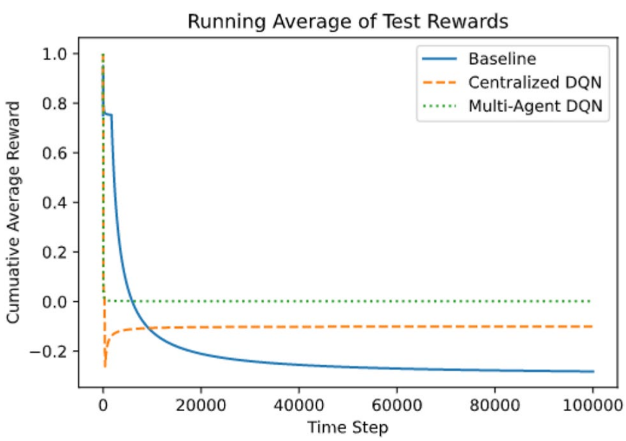


Fig. 11 Comparison of the running average received rewards

suggests that the reward function did encourage the desired behaviour from the nodes.

4 Experimental Results

In this section experiment design and obtained results are presented. Antenna test results are presented first, followed by overall system verification results.

4.1 Antenna Test Results

Each antenna was tested using the same methodology. Matching antenna were attached to the base station, and a separate TI launchpad. This launch pad was programmed to transmit one hundred packets of random characters at multiples of fifty feet distances to a final three hundred foot range. The base station, consisting of another TI launchpad and a laptop computer, recorded each incoming packet's RSSI at each distance (Fig. 12).

Antenna 3, the compact helical antenna will be the antenna used in future designs. It outperformed the current built-in antenna, and while it was slightly under performing as compared to the orthogonal compact PCB antennas, the slight increase in performance did not warrant the dramatic increase in antenna size.

4.2 Sensor Calibration with Machine Learning

The system's sensors were calibrated using a machine learning algorithm to correct the hysteresis in the sensor response. To do this, data was collected using a Campbell Scientific data logger with a CS650 Campbell Scientific soil moisture sensor. Calibration was done following [48, 49] using Gaussian Process Regression, Random Forest Regression, and Support Vector Regression.

The soil for the experiment was dried in an oven at 225 °F (107 °C) for 1 h. The data was collected by placing the two sensors in the same soil approximately 2 cm apart. Water was then added to the soil in increments of 15 ml every five minutes until 300 ml was added. Both sensors were sampled once per second. The collected data was then divided into training (90%) and test data (10%).

A comparison of the base sensor data and the true values can be seen in Fig. 13. After calibration, the sensor values closely tracked the true values, as shown in Fig. 14. All values assume the Campbell Scientific data logger values as the ground truth values. Computations were completed using methods described in [50, 51, 52].

4.3 System Validation Results

Based on the test data from the experiment, we see close agreement between the sensor systems with the proposed system explaining over 97% of the variance in the Campbell Scientific system's measurements.

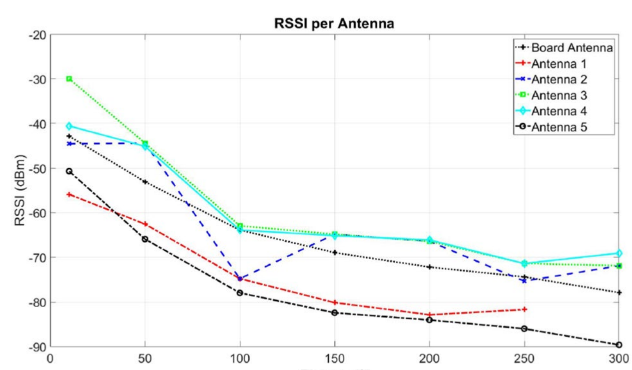


Fig. 12 RSSI of each antenna at each distance measured

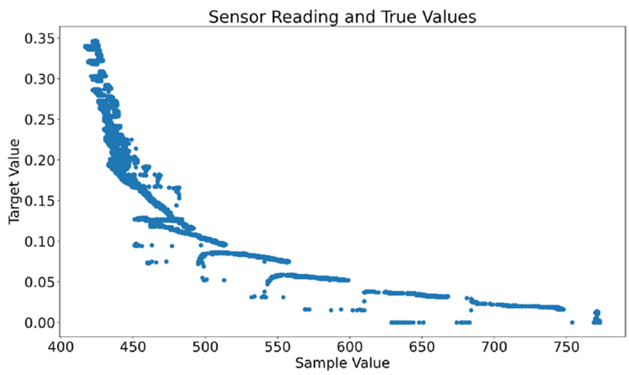


Fig. 13 Target and observed soil moisture values before processing

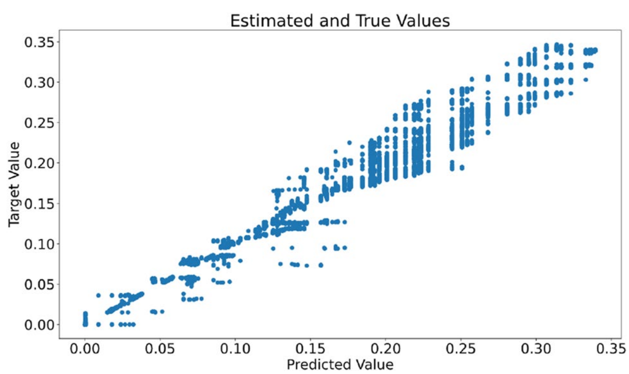


Fig. 14 Training targets and training data predicted values

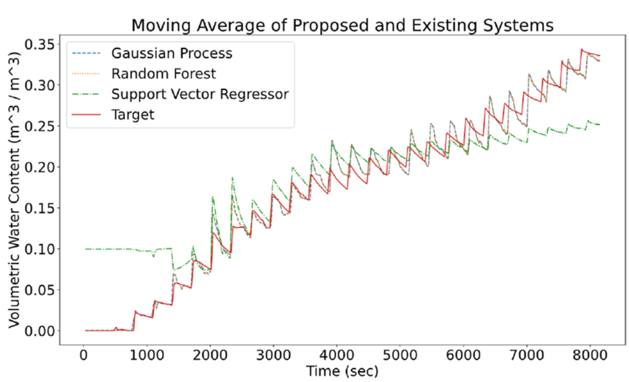


Fig. 15 A comparison of machine learning models used to calibrate the system

Of the variety of model types considered for the system calibration, Random Forest and Gaussian Process models provided similar results that were an excellent match to the target data. Both provided an r^2 value of approximately 0.972, with the random forest model slightly higher. A view of the results from the calibration are shown in

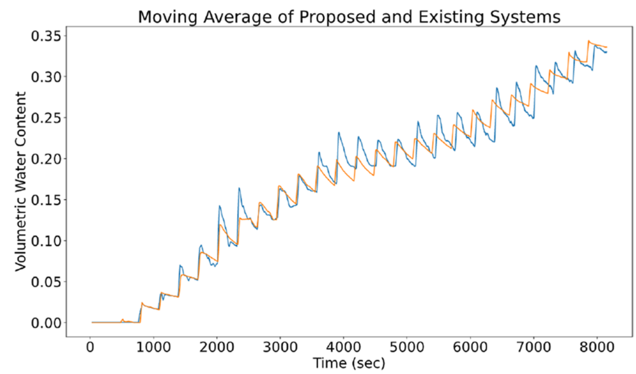


Fig. 16 Calibration results on combined test and training data

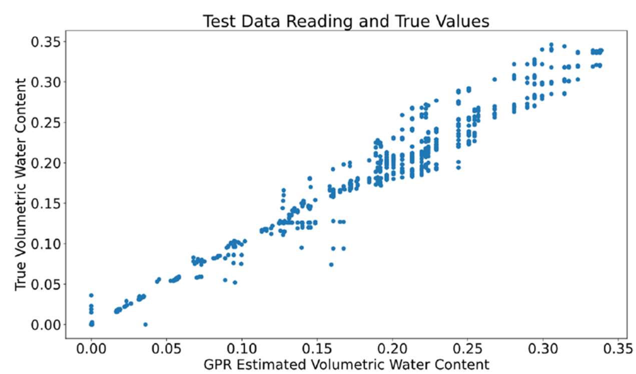


Fig. 17 Test targets and test data predicted values for the Gaussian Process Regression model

Fig. 15. The random forest and Gaussian process models provide almost identical estimates of the target signal. The support vector regression model offers inferior performance to the other methods. A variety of kernels were used in the models, and radial basis function kernels consistently outperformed the other options (Fig. 16).

A comparison of the test data true and predicted results is shown in Fig. 17. It is worth noting that the variance in the prediction increases with larger volumetric water content readings, likely due to hysteresis. Some variance may also be due to slight differences in sampling frequency as the proposed system does not have a real time clock to maintain an exact sampling rate. Since the system will sample every 15 minutes at deployment, we do not expect this to be an issue.

Once the data were cleaned, results were compared using a variety of metrics. These measures are in Table 3. A visual comparison of the data can be seen in Fig. 17. Based on this experiment, we see that the systems provide a comparable set of measurements.

Table 3 Test metrics for the calibration models

	Explained variance	MSE	MAE
Random forest	0.9721	0.0003	0.0108
Gaussian process	0.9718	0.0003	0.0109
Support vector	0.7501	0.0024	0.0373
No calibration	− 1.3e6	2.7e5	502.36

5 Concluding Remarks

In this paper, a low cost and reliable wireless soil moisture sensing system is proposed to enable high spatio-temporal data collection for improving our understanding of forest ecosystems. The developed methods allow for implementation of low-power sensor networks with optimized control using artificial intelligence and machine learning. This control system is comparable to the round-robin style baseline in terms of network control, and offers stable throughput in the network and conserves power. Furthermore, there is scope to extend the reinforcement learning algorithm to more nodes and other network architectures. The multi-agent approach offers power consumption benefits compared to the baseline used, and makes the best use of available network resources. In addition to these simulation results, we also compared the proposed system with industry standard wired systems in a field experiment. The results show reasonably similar data at much lower cost.

Future work includes enhancing the the sensor node with additional sensor types (soil and ambient temperature, snow depth, and more) and scaling up the network with more sensor nodes [53, 54, 55].

Acknowledgements The authors would like to thank our sponsor, National Science Foundation (NSF) EPSCoR Track 2 program, and our collaborators at the University of New Hampshire and University of Vermont.

Funding This project was financially sponsored by National Science Foundation (NSF) Grant No. 1920908.

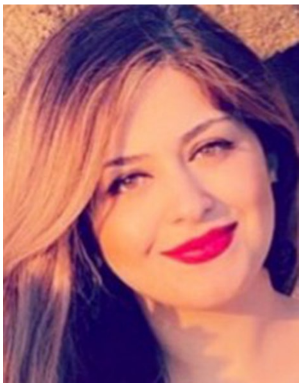
References

1. R. Kumar Dwivedi, S. Pandey and R. Kumar, A Study on Machine Learning Approaches for Outlier Detection in Wireless Sensor Network, in *2018 8th International Conference on Cloud Computing, Data Science Engineering (Confluence)*, Noida, pp. 189–192, 2018.
2. S. Khosroazad, S. Naderi and A. Abedi, Using Physical Layer Network Coding to Improve NOMA System Throughput with Energy Harvesting Users, in *2019 IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA*, pp. 1–6, 2019.
3. W. Guo, L. Hao and W. Di, Application and Development of Artificial Intelligence Technology for the Data Management and Analysis in Forestry, in *2009 International Conference on Artificial Intelligence and Computational Intelligence*, Shanghai, pp. 438–441, 2009. https://doi.org/10.1109/AICI.2009.12.
4. Y. Liu, B. Xiao and P. Liu, Constructing a Crisis Response System Using Artificial Intelligent Strategy, in *2009 International Conference on Networking and Digital Society*, Guiyang, Guizhou, pp. 129–132, 2009. https://doi.org/10.1109/ICNDS.2009.38.
5. D. L. P. Correia, W. Bouachir, D. Gervais, D. Pureswaran, D. D. Kneeshaw and L. De Grandpre, Leveraging artificial intelligence for large-scale plant phenology studies from noisy time-lapse images, *IEEE Access*, Vol. 8, pp. 13151–13160, 2020. https://doi.org/10.1109/ACCESS.2020.2965462.
6. J. Yang, P. Huang, F. Dai, Y. Sun, L. Wang and H. Bi, Application of Deep Learning in Wood Classification, in *2019 IEEE International Conference on Computer Science and Educational Informatization (CSEI)*, Kunming, China, pp. 124–129, 2019. https://doi.org/10.1109/CSEI47661.2019.8938960
7. J. Li and R. Liu, Applying data mining to forest maturity forecasting, in *2008 3rd International Conference on Intelligent System and Knowledge Engineering*, Xiamen, pp. 351–353, 2008. https://doi.org/10.1109/ISKE.2008.4730954.
8. R. Husain and R. Vohra, A survey on machine learning in wireless sensor networks, *International Education Research Journal (IERJ)*, Vol. 3, No. 1, pp. 17–18, 2017.
9. S. I. Seneviratne, et al., Investigating soil moisture-climate interactions in a changing climate: a review, *Earth-Science Reviews*, Vol. 99, No. 3–4, pp. 125–161, 2010.
10. T. Whitney, T. Nicholas, S. Naderi and A. Abedi, A low cost power efficient wireless soil moisture sensor network for forest ecosystem monitoring, *IEEE MIT Undergraduate Research Technology Conference (URTC)*, Vol. 2020, pp. 1–4, 2020.
11. R. D. Koster, et al., Regions of strong coupling between soil moisture and precipitation, *Science*, Vol. 305, No. 5687, pp. 1138–1140, 2004.
12. D. R. Legates, et al., Soil moisture: a central and unifying theme in physical geography, *Progress in Physical Geography: Earth and Environment*, Vol. 35, No. 1, pp. 65–86, 2011.
13. J. K. Green, S. I. Seneviratne, A. M. Berg, K. L. Findell, S. Hagemann, D. M. Lawrence and P. Gentine, Large influence of soil moisture on long-term terrestrial carbon uptake, *Nature*, Vol. 565, No. 7740, pp. 476–479, 2019.
14. J. Gornall, R. Betts, E. Burke, R. Clark, J. Camp, K. Willett and A. Wiltshire, Implications of climate change for agricultural productivity in the early twenty-first century, *Philosophical Transactions of the Royal Society B: Biological Sciences*, Vol. 365, No. 1554, pp. 2973–2989, 2010.
15. T. E. Ochsner, et al., State of the art in large-scale soil moisture monitoring, *Soil Science Society of America Journal*, Vol. 77, No. 6, pp. 1888–1919, 2013.
16. G. L. Schaefer, M. H. Cosh and T. J. Jackson, The USDA natural resources conservation service soil climate analysis network (SCAN), *Journal of Atmospheric and Oceanic Technology*, Vol. 24, No. 12, pp. 2073–2077, 2007.
17. J. E. Bell, et al., U.S. climate reference network soil moisture and temperature observations, *Journal of Hydrometeorology*, Vol. 14, No. 3, pp. 977–988, 2013.
18. W. A. Dorigo, et al., The international soil moisture network: a data hosting facility for global in situ soil moisture measurements, *Hydrology and Earth System Sciences*, Vol. 15, No. 5, pp. 1675–1698, 2011.
19. N. B. A. Karim and I. B. Ismail, *Soil Moisture Detection Using Electrical Capacitance Tomography (ECT) Sensor*, Perak Darul Ridzuan, Malaysia, 2011.
20. S. Naderi, S. Khosroazad and A. Abedi, Relay-assisted wireless energy transfer for efficient spectrum sharing in harsh

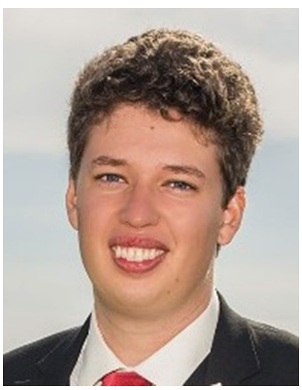
environments, *International Journal of Wireless Information Networks (IJWIN)*, Vol. 153, pp. 1–10, 2022.
21. D. Binkley and R. F. Fisher, *Ecology and Management of Forest Soils*, Wiley-Blackwell, Oxford, 2013.
22. A. Samouelian, I. Cousin, A. Tabbagh, A. Bruand and G. Richard, Electrical resistivity survey in soil science: a review, *Soil and Tillage Research*, Vol. 83, pp. 173–193, 2005.
23. D. A. Robinson, C. S. Campbell, J. W. Hopmans, B. K. Hornbuckle, S. B. Jones, R. Knight, F. Ogden, J. Selker and O. Wendroth, Soil moisture measurements for ecological and hydrological watershed scale observatories: A review, *Vadose Zone Journal*, Vol. 7, pp. 358–389, 2008.
24. W. W. Verstraeten, F. Veroustraete, C. J. Van der Sande, I. Grootaers and J. Feyen, Soil moisture retrieval using thermal inertia, determined with visible and thermal space borne data, validated for European forests, *Remote Sensing of Environment*, Vol. 101, pp. 299–314, 2006.
25. R. Sugiura, N. Noguchi and K. Ishii, Correction of low-altitude thermal images applied to estimating soil water status, *Biosystems Engineering*, Vol. 96, pp. 301–313, 2007.
26. J. P. Walker, G. R. Willgoose and J. D. Kalma, In situ measurement of soil moisture: a comparison of techniques, *Journal of Hydrology*, Vol. 293, pp. 85–99, 2004.
27. O. Merlin, J. P. Walker, R. Panciera, R. Young, J. Kalma and E. J. Kim, Soil Moisture Measurement in Heterogeneous Terrain, in *MODSIM 2007 International Congress on Modelling and Simulation*, 2007.
28. W. R. Belisle, A. Sharma and T. L. Coleman, An optical reflectance technique for soil moisture measurement. I. Theory, description, and application, in *IGARSS '96. 1996 International Geoscience and Remote Sensing Symposium*, Lincoln, NE, USA, Vol. 2, pp. 1315–1319, 1996.
29. V. S. Palaparthy, S. Lekshmi, J. John, S. Sarik, M. Sh. Baghini and D. N. Singh, Soil Moisture Measurement System for DPHP Sensors and In Situ Applications, in *Proceedings 4th International Symposium on Electronic System Design*, pp. 12–15, 2013.
30. O. Calla, D. Bohra, R. Vyas, B. Purohit, R. Prasher, A. Loomba and N. Kumar, Measurement of soil moisture using microwave radiometer, in *2008 International Conference on Recent Advances in Microwave Theory and Applications*, Jaipur, pp. 621–624, 2008.
31. C. K. Sahu and P. Behera, "A low cost smart irrigation control system," in 2015 2nd International Conference on Electronics and Communication Systems (ICECS), Coimbatore, pp. 1146–1152, 2015.
32. G. J. Gaskin and J. D. Miller, Measurement of soil water content using a simplified impedance measuring technique, *Journal of Agricultural Engineering Research*, Vol. 63, pp. 153–160, 1996.
33. A. Fares, H. Hamdhani and D. M. Jenkias, Temperature-dependent sealed frequency: Improved accuracy of multisensory capacitance probes, *Soil Science Society of America Journal*, Vol. 71, pp. 894–900, 2007.
34. E. Veldkamp and J. J. O'Brien, Calibration of a frequency domain reflectometry sensor for humid tropical soils of volcanic origin, *Soil Science Society of America Journal*, Vol. 64, No. 5, pp. 1549–1553, 2000.
35. W. Kunzler, S. G. Calvert and M. Laylor, Measuring Humidity and Moisture with Fiber Optic Sensors, in *Proceedings of Sixth Pacific Northwest Fiber Optic Sensor Workshop (SPIE)*, Vol. 5278, 2003.
36. S. K. Khijwania, K. L. Srinivasanb and J. P. Singha, An evanescentwave optical fiber relative humidity sensor with enhanced sensitivity, *Journal of Sensors and Actuators B: Chemical*, Vol. 104, No. 2, pp. 217–222, 2005.
37. S. Muto, A. Fukasawa, T. Ogawa, M. Morisawa and H. Ito, Optical detection of moisture in air and in soil using dye-doped plastic fibers, *Japanese Journal of Applied Physics*, Vol. 29, pp. L1023–L1025, 1990.
38. T. Seiyama, N. Yamazoe and H. Arai, Ceramic humidity sensors, *IEEE Transactions on Components Hybrids, and Manufacturing Technology*, Vol. 3, No. 2, pp. 85–96, 1980.
39. I. F. Long and B. K. French, Measurement of soil moisture in the field by neutron moderation, *Journal of Soil Science*, Vol. 18, pp. 149–166, 2006.
40. R. Wallace, CC-Antenna-DK2 and Antenna Measurements Summary, in *Texas Instruments*, Dallas, TX, USA, October 2017, May 2021, https://www.ti.com/lit/an/swra496a/swra496a.pdf
41. A. Jeerige, D. Bein and A. Verma, Comparison of Deep Reinforcement Learning Approaches for Intelligent Game Playing, in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, USA, pp. 0366–0371, 2019. https://doi.org/10.1109/CCWC.2019.8666545.
42. C.-G. Li, M. Wang, Q.-N. Yuan, A multi-agent reinforcement learning using actor-critic methods, in *Proceedings of the 7th International Conference on Machine Learning and Cybernetics, ICMLC*. Vol. 2, pp. 878–882, 2008. https://doi.org/10.1109/ICMLC.2008.4620528.
43. J. Su, W. Liu, and K. Yue, A Network Routing Algorithm Based on the Coalitional Game Theory, in *2009 International Conference on Computational Intelligence and Natural Computing*, Wuhan, pp. 409–412, 2009. https://doi.org/10.1109/CINC.2009.252.
44. A. K. Charles and N. Pissinou, Mitigating selfish misbehavior in multi-hop networks using stochastic game theory, in *IEEE Local Computer Network Conference*, Denver, CO, pp. 232–235, 2010. https://doi.org/10.1109/LCN.2010.5735709.
45. W. Qiang and Z. Zhongli, Reinforcement learning model, algorithms and its application, in *2011 International Conference on Mechatronic Science, Electric Engineering and Computer (MEC)*, Jilin, pp. 1143–1146, 2011. https://doi.org/10.1109/MEC.2011.6025669.
46. D. J. Young and Norman Beaulieu, The generation of correlated Rayleigh random variates by inverse discrete Fourier trans form, *IEEE Transactions on Communications*, Vol. 48, pp. 1114–1127, 2000. https://doi.org/10.1109/26.855519.
47. D. Clevert, T. Unterthiner, and S. Hochreiter, Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). cs.LG 1511.07289, 2015.
48. S. Urban, M. Ludersdorfer and P. van der Smagt, Sensor calibration and hysteresis compensation with heteroscedastic Gaussian processes, *IEEE Sensors Journal*, Vol. 15, No. 11, pp. 6498–6506, 2015. https://doi.org/10.1109/JSEN.2015.2455814.
49. L. O. H. Wijeratne, D. R. Kiv, A. R. Aker, S. Talebi and D. J. Lary, Using machine learning for the calibration of airborne particulate sensors, *Sensors*, Vol. 20, No. 1, pp. 99, 2019. https://doi.org/10.3390/s20010099.
50. Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825–2830, 2011.
51. The pandas development team, pandas-dev/pandas: Pandas, Zenodo. 2020. https://doi.org/10.5281/zenodo.3509134.
52. W. McKinney, Data Structures for Statistical Computing, in *Proceedings of the 9th Python in Science Conference*, pp.56–61, 2010. https://doi.org/10.25080/Majora-92bf1922-00a
53. USDA Forest Service, Northern Research Station. 2019. Hubbard Brook Experimental Forest: 15 Minute Solar Radiation Measurements, 2014 - present ver 1. Environmental Data Initiative. https://doi.org/10.6073/pasta/22a42fd1aa4f8e935db50837de4893cb Accessed 2020-07-29
54. F. R. Hampel, The influence curve and its role in robust estimation, *Journal of the American Statistical Association*, Vol. 69, pp. 382–393, 1974.

55. D. Binkley and R. F. Fisher, *Ecology and Management of Forest Soils*, Wiley-Blackwell, Oxford, 2013.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Sonia Naderi is a PhD students at the Electrical and Computer Engineering department, University of Maine, Orono. Her research involves wireless sensor networks for environmental monitoring.



Kenneth Bundy received his BA (Honors) and MS in Mathematics from University of Maine in 2014, 2018. He has worked at the Electrical and Computer Engineering department, Wireless Sensor Networks (WiSeNet) laboratory as research assistant during his graduate studies and have published several papers in IEEE conferences.

Thayer Whitney is a graduate student in the Electrical and Computer Engineering department at the University of Maine, Orono. His research currently focuses on radio communications, low power systems, and sensor networks.

Ali Abedi received his B.Sc and M.Sc degrees in Electrical Engineering from Sharif University of Technology and his PhD in Electrical and Computer Engineering, from University of Waterloo in 1996, 1998, and 2004, respectively. Dr. Abedi joined the University of Maine, Orono in 2005, where he is currently Professor of Electrical and Computer Engineering and Associate Vice President for Research. He was visiting Associate Professor at the University of Maryland, College Park and Guest Researcher at NIST in 2012 and Faculty Fellow at NASA MSFC in 2016.

Aaron Weiskittel received his B.Sc in Natural Resources from The Ohio State University and his M.Sc and PhD in Forestry from Oregon State University. Dr. Weiskittel joined the University of Maine, Orono in 2008, where he is currently Professor of Forest Biometrics and Modeling, the Irving Chair of Forest Ecosystem Management, and Director of the Center for Research on Sustainable Forests. He is also Director of the National Science Foundation's Center for Advanced Forestry Systems. Dr. Weiskittel currently serves as an associate editor for Canadian Journal of Forest Research, Forest Ecology & Management, and Frontiers in Forests and Global Change.

Alexandra Contosta received her BA in Philosophy from Villanova University in 1998, her MS in Conservation Biology from Antioch University in 2005 and her PhD in Earth and Environmental Science from the University of New Hampshire in 2011, where she is currently Research Assistant Professor in the Earth Systems Research Center. Dr. Contosta has also held appointments as Visiting Research Fellow at the National Socio-Environmental Synthesis Center (SESYNC) and Public Engagement Fellow with the Hubbard Brook research Foundation.