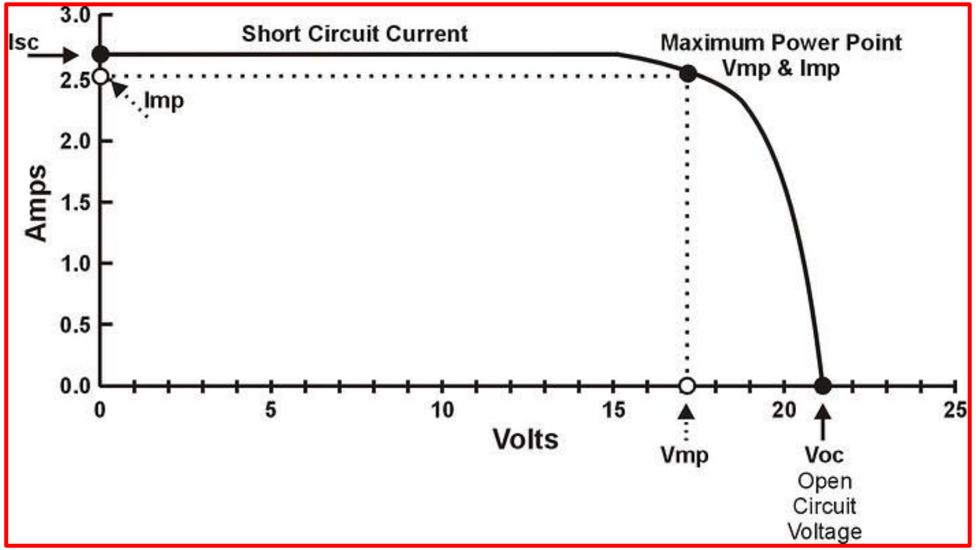
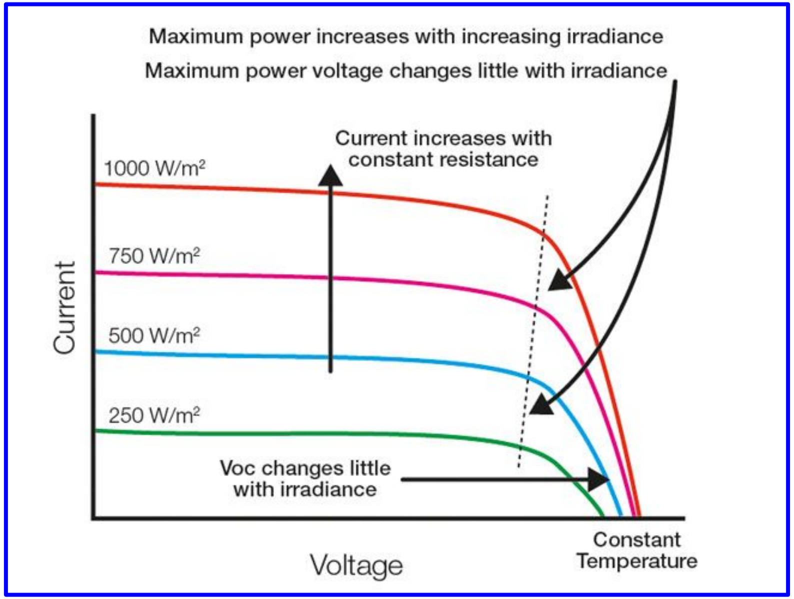
AI Driven Maximum Power Point Tracking - A Comparison Study

Basheer Ammar, Morgan Williams, Selasi Etchey

Background

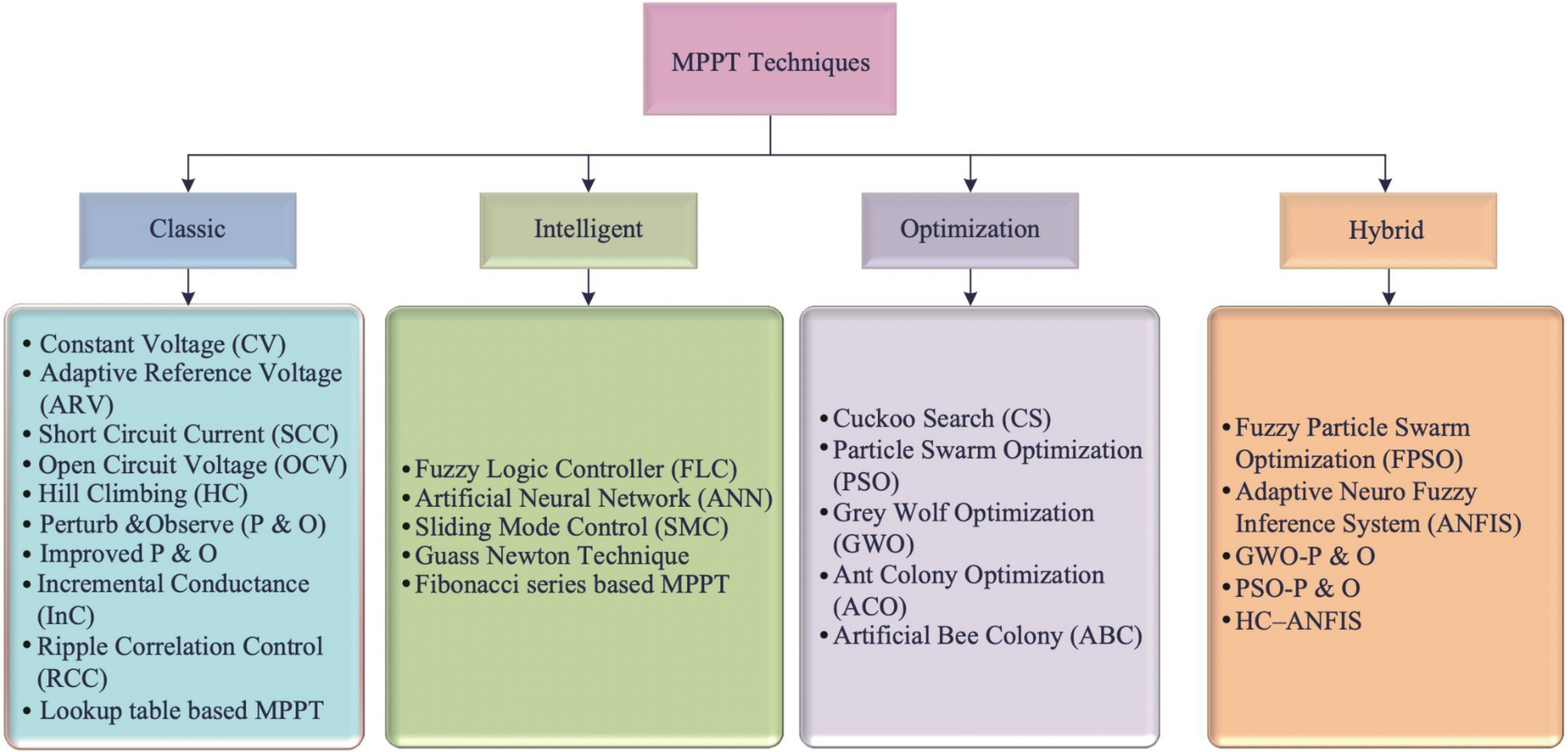




P-V Curve: ***as voltage varies so does the current***

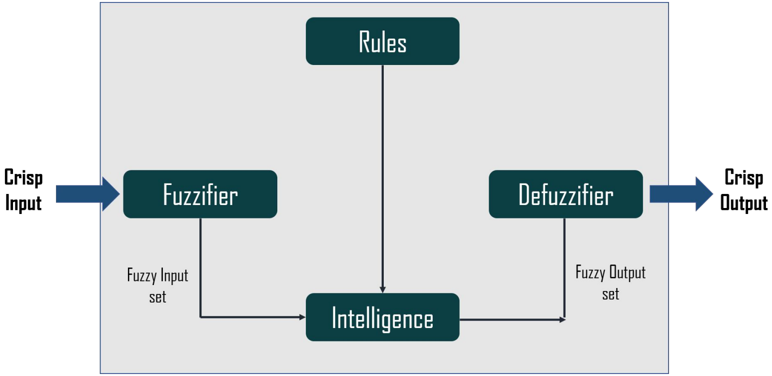
I-V Curve: ***varies as the irradiance changes***

Project Motivation



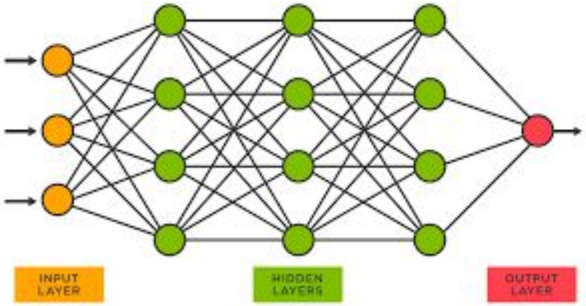
The 4 Categories of Maximum Power Point Tracking Algorithms

Referenced Works [Fuzzy Logic]



- **Fuzzy Logic**
  - Yilmaz, Unal, et al. “PV System Fuzzy Logic MPPT Method and Pi Control as a Charge Controller.” *Renewable and Sustainable Energy Reviews*, vol. 81, 2018, pp. 994–1001., https://doi.org/10.1016/j.rser.2017.08.048.
    - The inspiration for the setup of our fuzzy logic controller, and a previous attempt at using such a controller for the purpose of MPPT. Takes in error and change of error as an input, and returns a duty cycle output. Overall system explained also employs photovoltaic panels and a Proportional Integral (PI) controller operating a buck converter.
  - “Fuzzy Logic Toolbox Documentation.” *MathWorks*, https://www.mathworks.com/help/fuzzy/index.html?s_tid=CRUX_topnav.
    - Details the operation of the Fuzzy Logic Toolbox within MATLAB that was used to set up the environment used to conduct the fuzzy logic approach experiments.

Referenced Works [Neural Network]



- **Neural Network**
  - A. Laudani, F. R. Fulginei, A. Salvini, G. M. Lozito and F. Mancilla-David, "Implementation of a neural MPPT algorithm on a low-cost 8-bit microcontroller," *2014 International Symposium on Power Electronics, Electrical Drives, Automation and Motion*, 2014, pp. 977-981, doi: 10.1109/SPEEDAM.2014.6872101.
    - Implementation of a Neural Network based MPPT Algorithm on an 8-bit microcontroller. Method uses sensor data (Voltage,Temperature, and Current) in order to estimate incident irradiance on a solar panel as well as predicting optimal operating voltage of the panel.
  - F. Mancilla-David, F. Riganti-Fulginei, A. Laudani and A. Salvini, "A Neural Network-Based Low-Cost Solar Irradiance Sensor," in *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 3, pp. 583-591, March 2014, doi: 10.1109/TIM.2013.2282005.
    - Implementation of a neural network as a low-cost solar array irradiance estimator, bypassing the need for expensive pyranometers.This Irradiance sensor can be used to for more efficient control of large solar plants, where shading effects are unpredictable and widespread

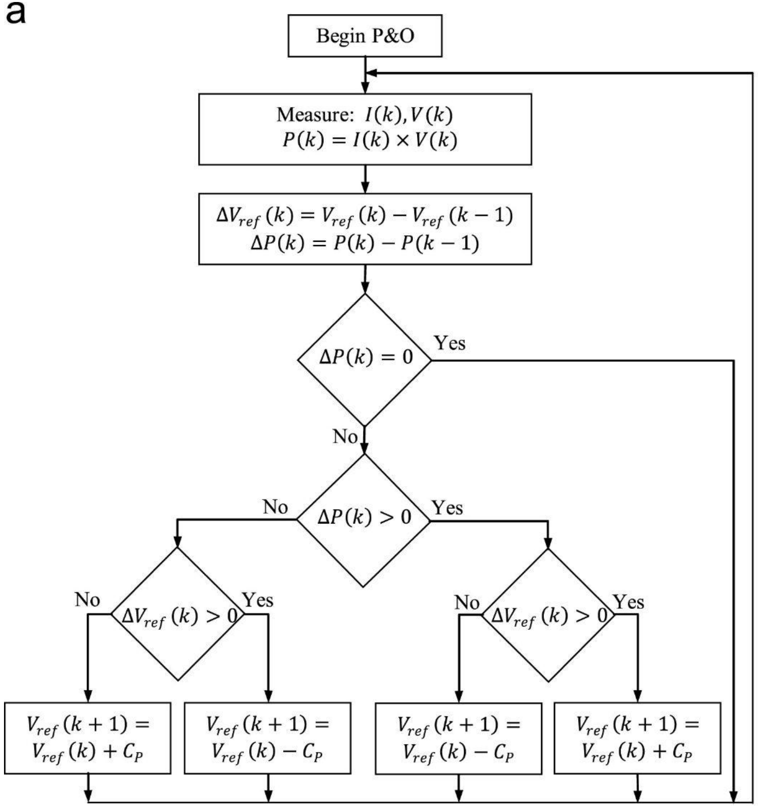
High Level Control Algorithm Testing Methodology

- Testing Methods
  - Under comparable irradiance and temperature conditions, start control algorithms at known setpoint (large or small duty cycle)
    - Measure time to reach steady state value [Convergence Time]
    - Using P&O Maximum Power Point as a reference, calculate discrepancies from Neural Network and Fuzzy Logic approaches [Convergence Accuracy]
    - Repeat under partially shaded conditions, observe maximum power value of each method and compare [Performance in Partial Shading]

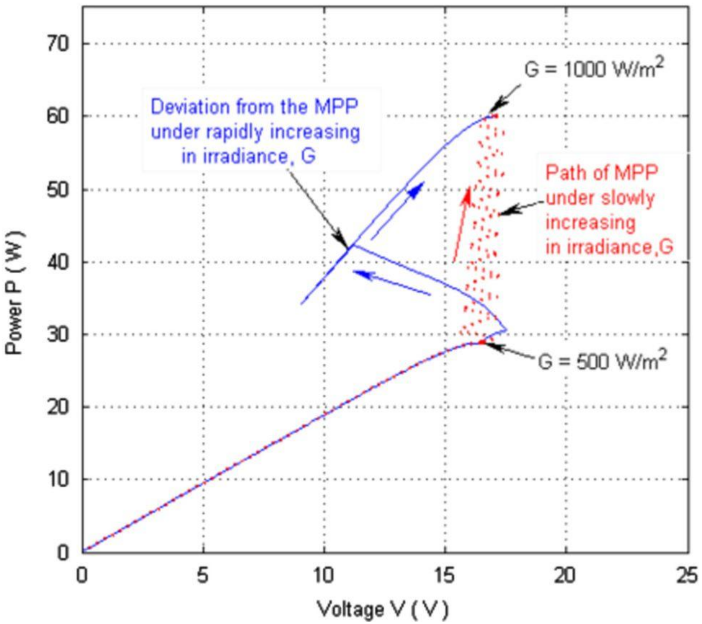
Project Approach: Perturb & Observe

Approach: Conventional MPPT Technique (Perturb & Observe)

a



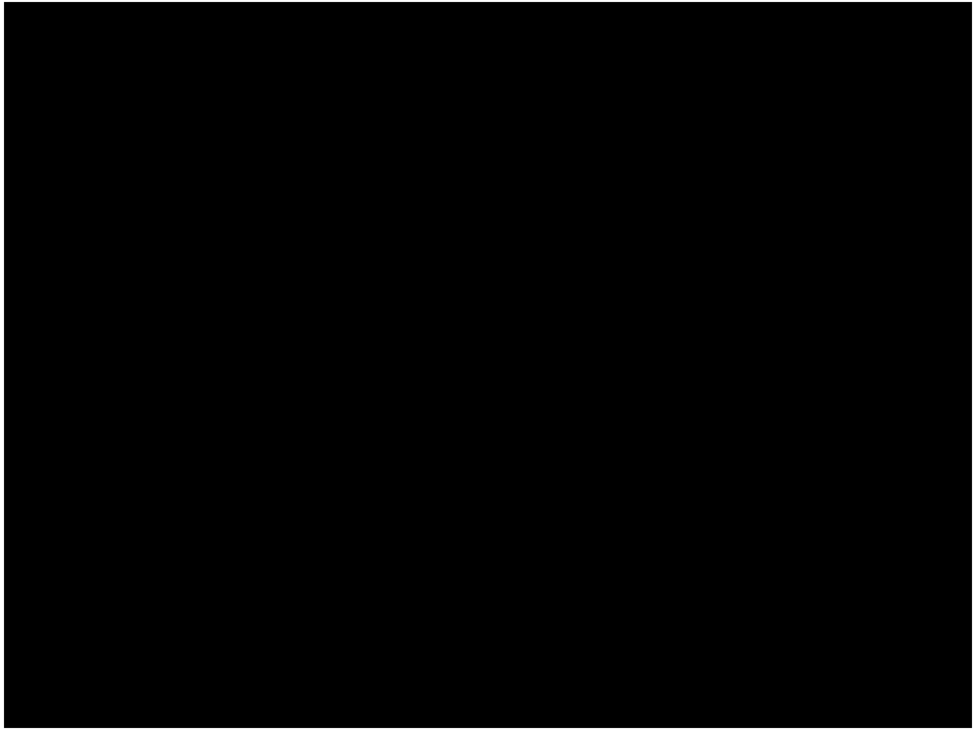
b



(a) Flow chart of the P&O MPPT algorithm (C_P is the perturbation step width); (b) path of MPP with the P&O algorithm under slowly (dotted line) and rapidly (continued line) changing irradiance.

B. Bendib, H. Belmili, F. Krim. A survey of the most used MPPT methods: conventional and advanced algorithms applied for photovoltaic systems. Renew. Sustain. Energy Rev., 45 (2015), 10.1016/j.rser.2015.02.009

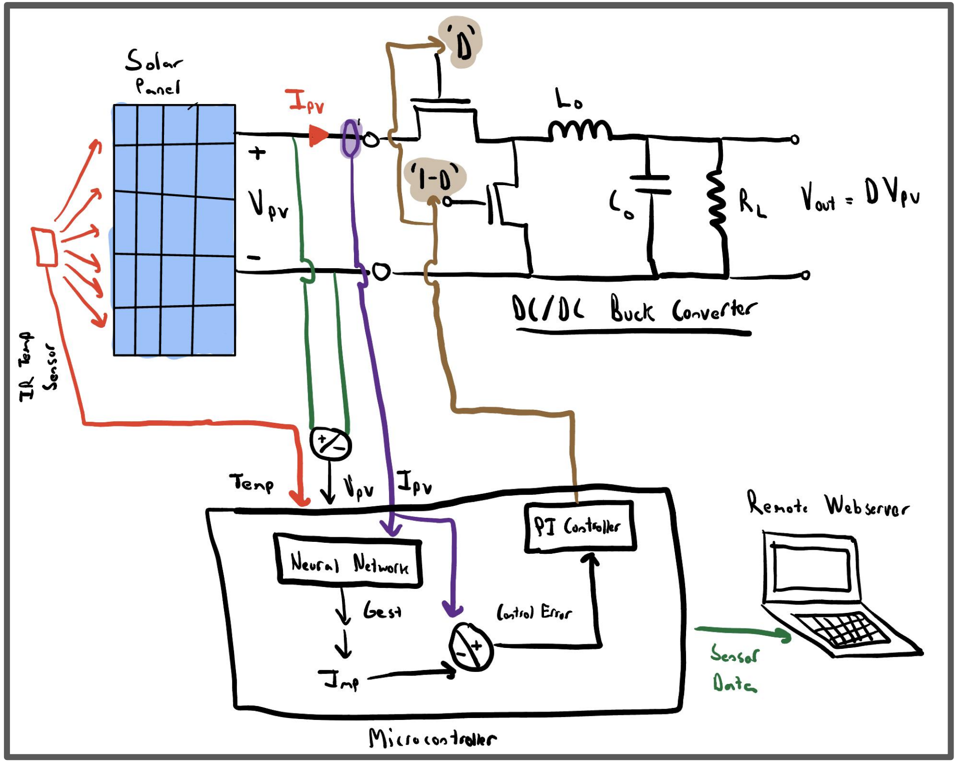
Experimentation and Results: P&O Algorithm



Perturb & Observe Test (No Shade and Partial Shade)

Project Approach: Neural Network

Neural Network Design Architecture

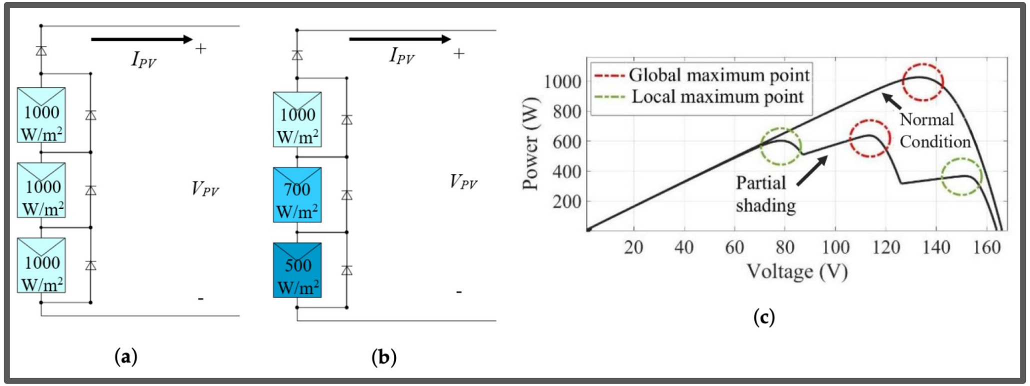


Example System Diagram

- Sensor Data from PV array (Voltage,Current,Temp) feed into a neural network that estimates irradiance
- Irradiance Estimate is used to calculate Maximum Power Point Current
- Maximum Power Point Current is compared against sensed current in a PI controller, and the system’s operating point is adjusted
- Sensor Data is broadcasted to web server for remote monitoring

Why Utilize a Neural Network for MPPT?

- Evaluates Maximum Power Point in a single step
  - Faster convergence when compared to conventional iterative step methods
  - Useful in compensating for large setpoint changes (abrupt shading occurrences)
- Results on less error in partial shading conditions
  - Partial shading on solar panels can result in a maximum power point curve that has multiple local peaks. Traditional MPPT algorithms can get stuck on these peaks



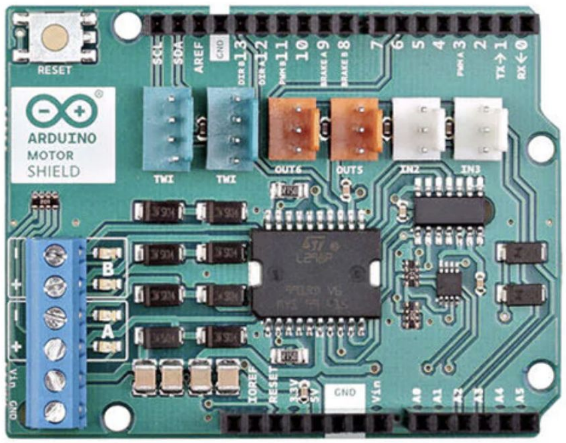
https://pdfs.semanticscholar.org/bfd8/f7c16147439ce75e8cf651659288c2e46ad0.pdf

Why Utilize a Neural Network for MPPT?

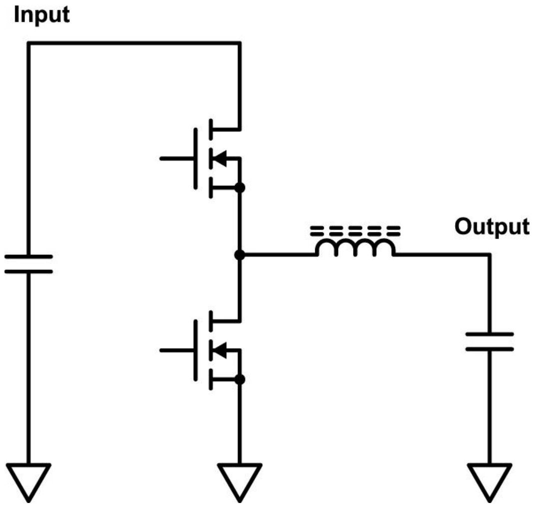
- Neural Network Irradiance Estimator can also be used as a low-cost pyranometer for efficient control of large solar farms
  - Dummy Solar Panel hooked up to irradiance estimator could be used for more precise tracking/control, and could account for panel inclination angles

Hardware Design: MPPT Regulator

- Switching DC - DC Converter
  - Using Full Bridge Motor Driver as a PWM DAC by low-pass filtering the output (In Closed loop operation, acts similar to Half Bridge Synchronous Buck Converter with body diode on the bottom leg FET)
    - Closed loop control is accomplished by linking PWM duty cycle input to control system output

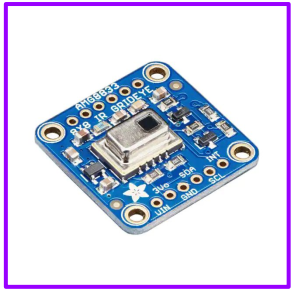


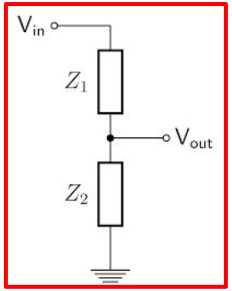
Arduino Motor Driver Shield

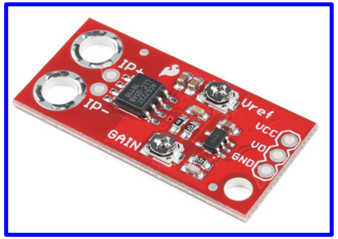


Half Bridge Synchronous Buck

Hardware Design: Sensors

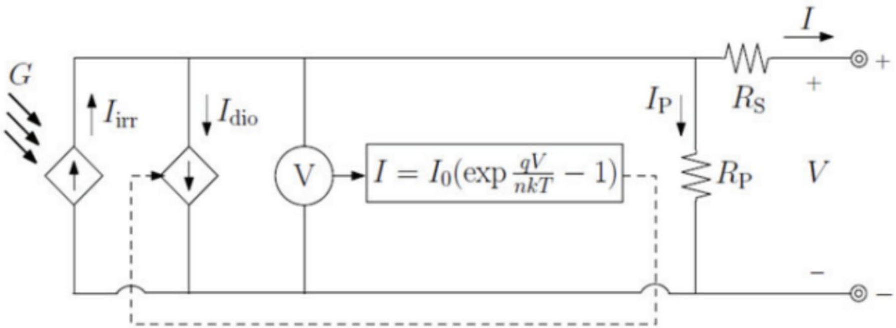




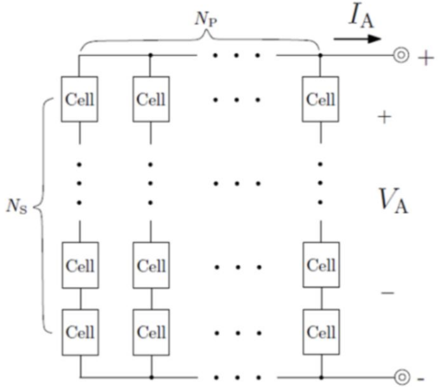


- Sensors
  - Adafruit AMG 8833 IR Thermal Camera
    - Pointed at Solar Panel and used to determine surface temperature of the panel
    - 8x8 grid temperature measurements are averaged across all pixels in the camera feed
  - Voltage Sensor
    - Simple Resistive Voltage Divider
    - Scales PV output voltage to limits of the Arduino Nano BLE Sense
  - Current Sensor
    - External Current Sensor for the Buck Converter (Half Bridge Motor Driver)
    - 0.8V/A is high enough resolution to sense the voltage coming out of the panels
      - Panel VOC ~ 13.2V and ISC ~ 0.8A

Software Design: PV Modelling



Equivalent Model of a Photovoltaic Cell



Physical Configuration of Photovoltaic Array

Mathematical Model Representation

$$I_A = I'_{IRR} - I'_0\left[\exp\left(\frac{q(V_A + I_A R'_S)}{N_S nkT}\right) - 1\right] - \frac{V_A + I_A R'_S}{R'_P}$$

$$\begin{aligned} I_{\mathrm{irr}} &= I_{\mathrm{irr,ref}}\frac{G}{G_{\mathrm{ref}}}(1 + \alpha'_T(T - T_{\mathrm{ref}})) \\ I_0 &= I_{0,\mathrm{ref}}\left(\frac{T}{T_{\mathrm{ref}}}\right)^3 \exp\left(\frac{E_{g,\mathrm{ref}}}{kT_{\mathrm{ref}}} - \frac{E_g}{kT}\right) \\ R_P &= R_{P,\mathrm{ref}}\left(\frac{G}{G_{\mathrm{ref}}}\right) \\ R_S &= R_{S,\mathrm{ref}} \\ n &= n_{\mathrm{ref}}. \end{aligned}$$

I-V Characteristics of a PV Array

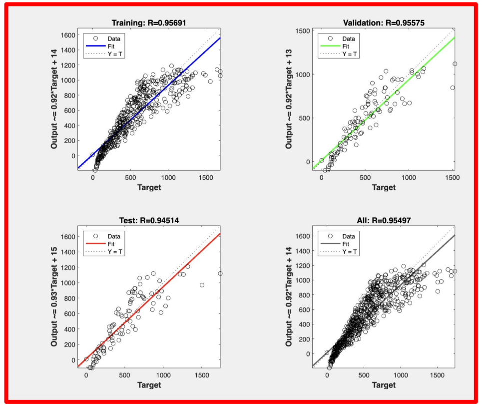
PV module parameter	*PV array parameter*
$N_P I_{IRR}$	I'_{IRR}
$N_P I_0$	I'_0
$N_S R_S/N_P$	R'_S
$N_S R_P/N_P$	R'_P

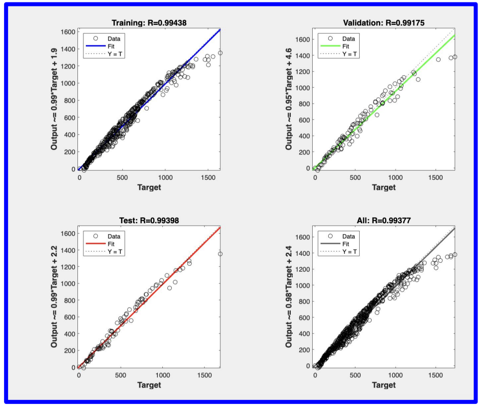
Scaling Parameters [Cell->Array]

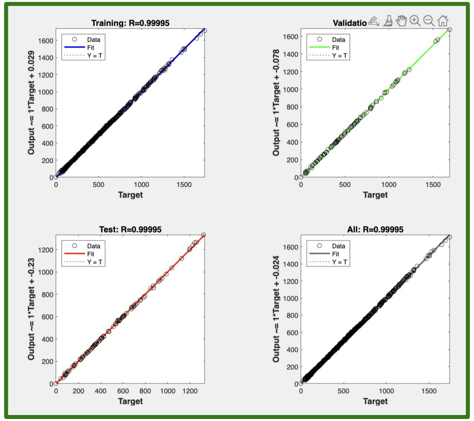
Software Design: Neural Network Control Implementation

- Neural Network Architecture and Training
  - 2 Layer Feedforward Network , 3 hidden layers, 70% Training Data, 15% Testing Data, 15% Validation Data
  - MATLAB script written to generate training data
    - 2000 data points generated from random variations of Voltage,Temperature,and Current

- PID Control Loop
  - Neural Network Irradiance estimate is used to calculate maximum power point current
    - Imp = Imp_ref *
  - Calculated Imp is compared with sensed PV output current, and the error signal is sent through a PI Controller to update the operating point of the voltage regulator







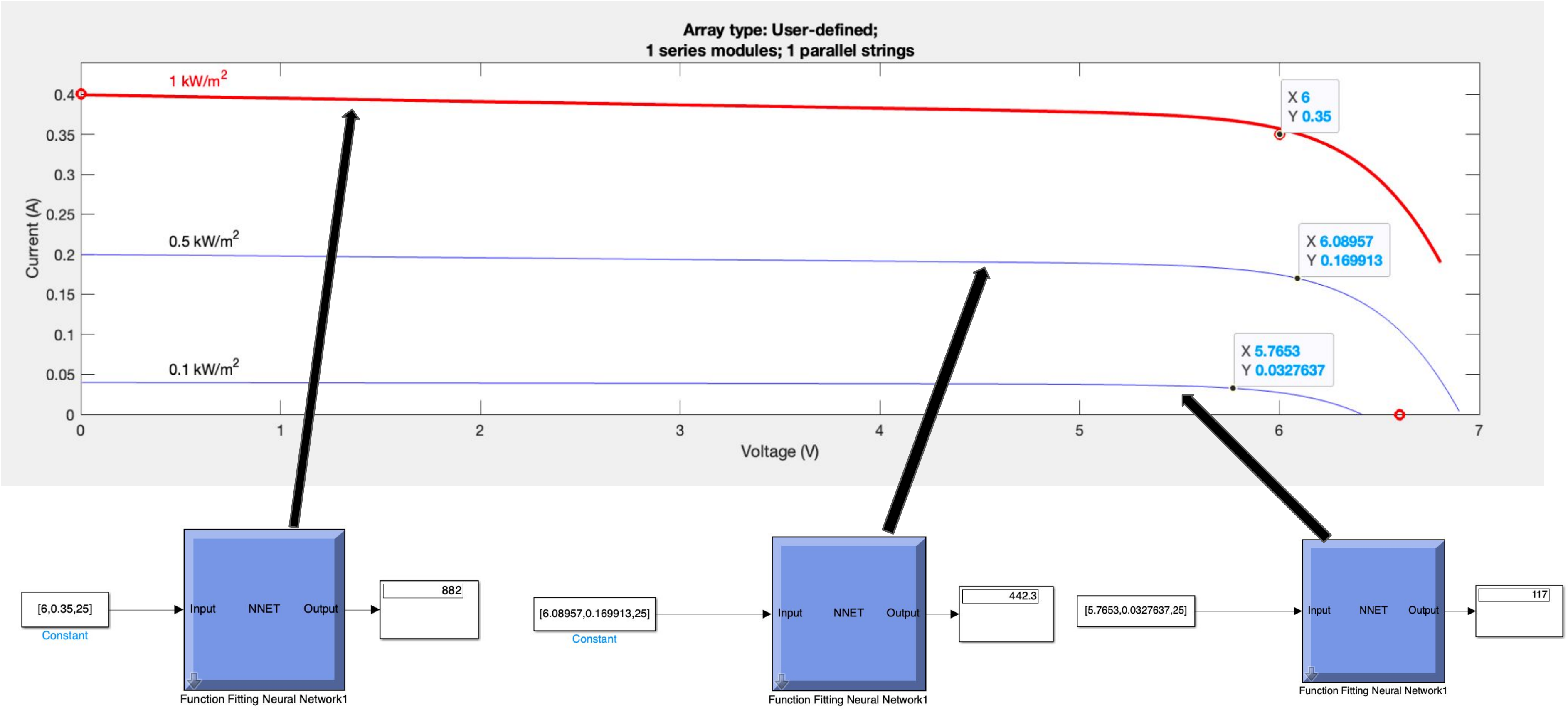
<u>Multivariate Fitting</u>

Top Left Image:
1 Hidden Layer

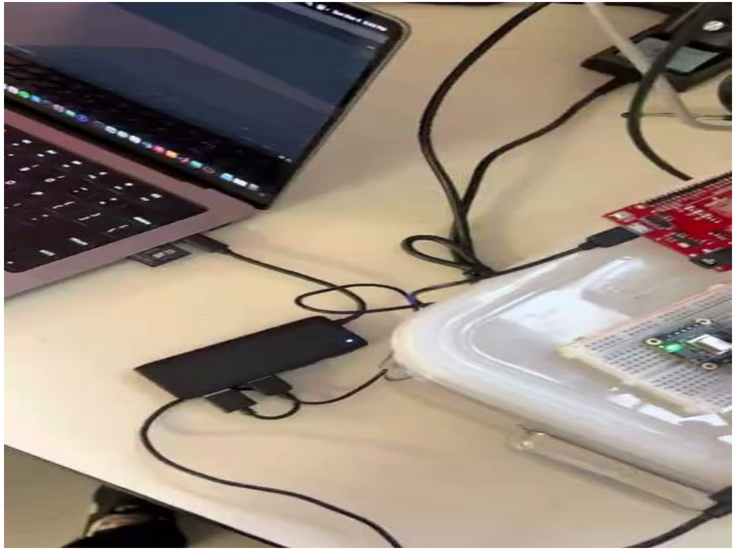
Top Right Image:
2 Hidden Layers

Bottom Image :
3 Hidden Layers

Approach: Neural Network



Experimentation and Results: Neural Network



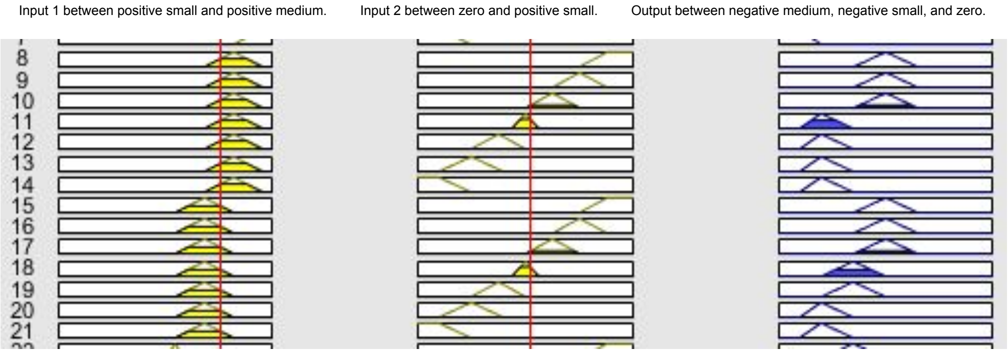
Open Loop Buck Converter Test

Simulink Irradiance Estimation Test

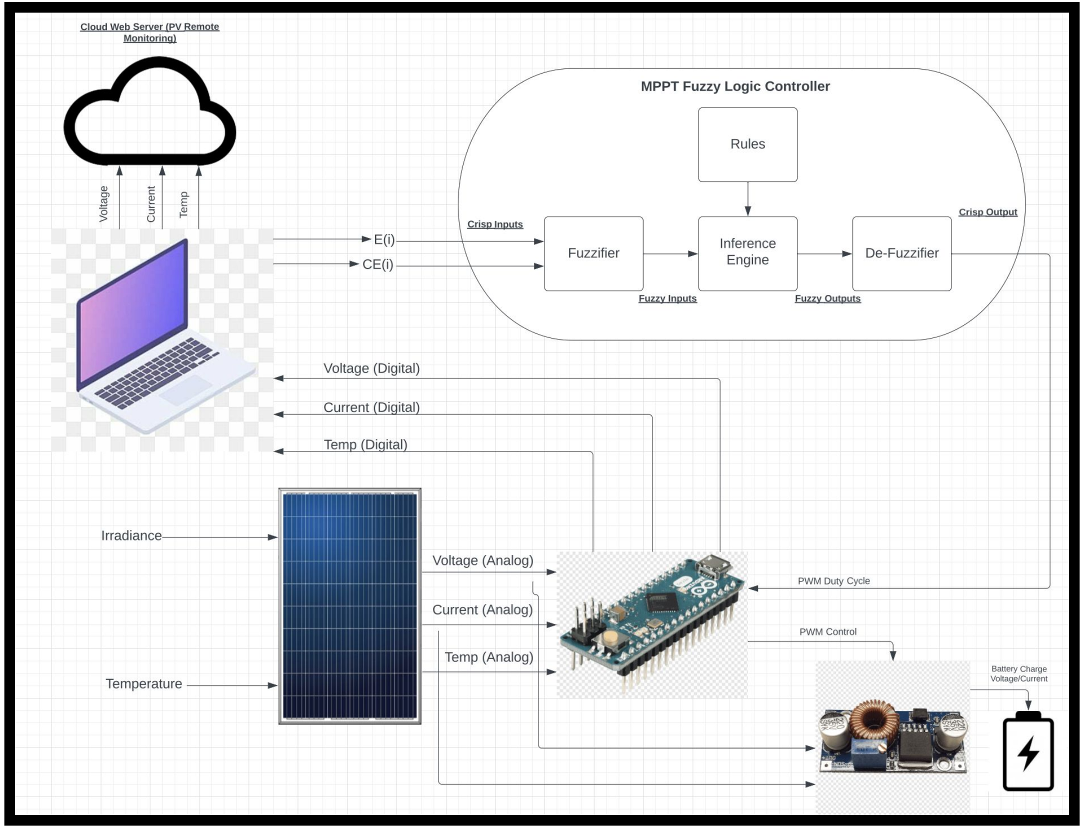
Project Approach: Fuzzy Logic

Approach: Fuzzy Logic - Tech Overview

- Logical model that determines outputs based on observations by determining the degree in which certain rules are met.
- Allows us to draw upon our knowledge of the meaning of observations in a way that goes beyond 1:1 mappings, makes sense of “in-between” cases using AI



Approach: Fuzzy Logic - Overall System Design

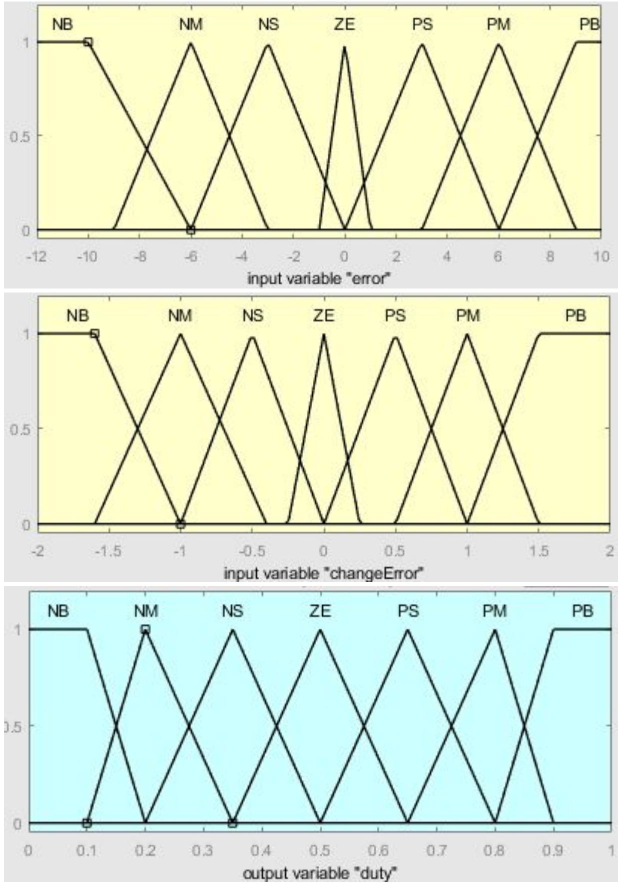


System Diagram.

Approach: Fuzzy Logic - FL Controller Philosophy

- Error: ΔP/ΔV
- Change in Error over one time step
- Adjustments to Yilmaz et al.'s method:
    - Narrower zero boundary for both inputs to match environmental expectations
    - Output from 0 to 1 to correspond to traditional duty cycle representation

Membership Functions.



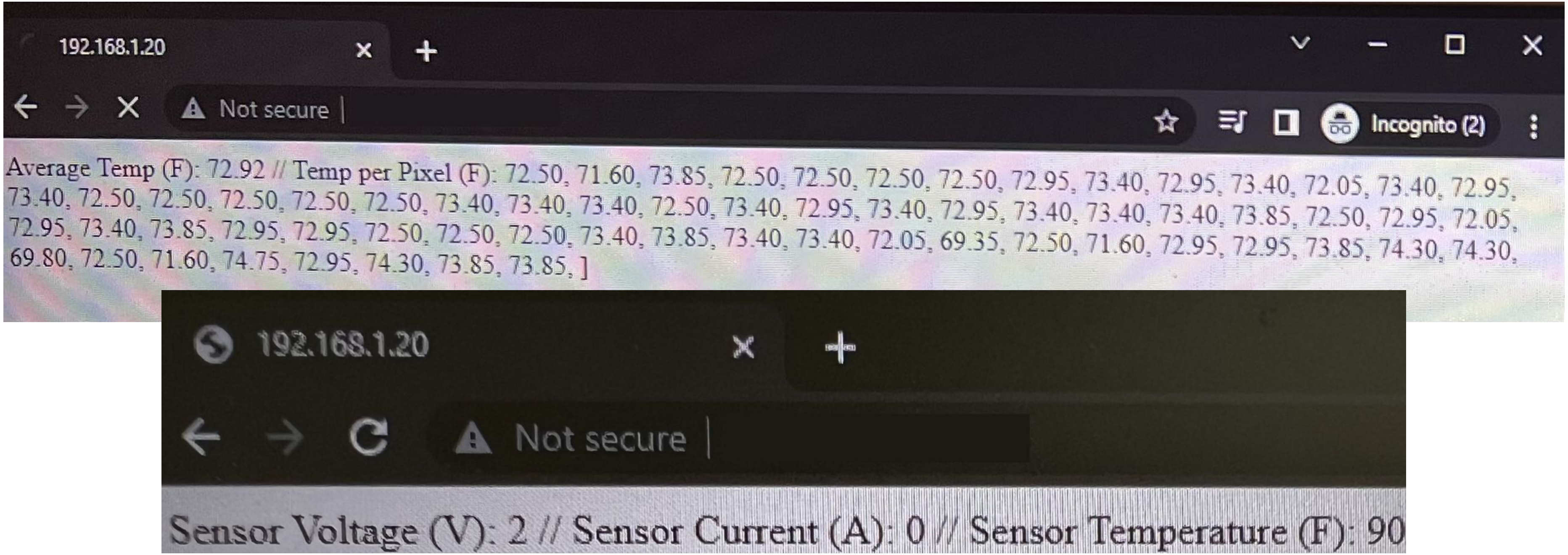
The rule table for FLC.

E/CE	PB	PM	PS	ZE	NS	NM	NB
PB	ZE	ZE	ZE	NB	NB	NB	NB
PM	ZE	ZE	ZE	NM	NM	NM	NM
PS	ZE	ZE	ZE	NS	NS	NM	NM
ZE	NS	NS	ZE	ZE	ZE	PS	PS
NS	PM	PM	PS	NS	ZE	PS	ZE
NM	PM	PM	PM	PB	ZE	ZE	NS
NB	PB	PM	PM	PB	ZE	ZE	ZE

Approach: MBED Real-Time Operating System

- We used the MBED RTOS to schedule threads in our Neural Network algorithm, Fuzzy Logic & Perturb and Observe
    - Three tasks:
        - Sensing the Current and Voltage Data
        - Sending data to the wifi server
        - Updating the system operating point
    - Better to have sensing and actuation running in parallel
    - Would better control the flow of data by treating them as shared resources

Experimentation and Results: Wi-Fi Data Transmission



- Performed using Arduino Nano RP2040 boards as a proof of concept

Obstacles and Pitfalls

- Hardware Issues and Setpoints
  - Initial buck design failed due to lack of gate driver to drive main power mosfet, so had to switch to ‘PWM DAC’ approach on short notice
  - Multiple sensor failures (current sensors) as well as a board failure (fried BLE Sense board while testing initial buck converter design)
- Real-time control system performance hindered by Matlab serial data link
  - No simultaneous Arduino and Simulink serial transmission possible
  - Trouble implementing neural network on microcontroller hardware, approach became to get the control working through MATLAB as a serial medium
- Lack of good weather conditions for hardware tests on PV panels
  - Numerous cloudy and rainy weather days in late November timeframe when testing was ready to be conducted, limited window for testing (8am - 4pm)
  - No access to lab instrumentation outside

Project Status

- Open Loop Objectives
  - Collecting and conditioning sensor data ✓
  - Wifi Transmission of sensor data ✓
  - Open Loop Regulator Control ✓
  - Open Loop Neural Network Irradiance Estimate Testing w/ MATLAB ✓
  - Open Loop Fuzzy Logic Inference System Testing w/ MATLAB ✓
  - MBED OS Integration ✗
- Closed Loop Objectives
  - Neural Network Closed Loop MPPT ✗
  - Fuzzy Logic Closed Loop MPPT ✗
  - P&O MPPT ✓
  - Closed Loop Control Comparison ✗

Project Improvements

- Hardware
  - Switch MPPT Regulator Design to standard non-synchronous (or synchronous) buck design with gate driver
    - Would reduce 6V minimum solar panel input voltage and possible aid in increasing converter output range
  - Design system on PCB to increase regulator efficiency and clean up wiring
  - Design around more sensitive current sensor (current measurements affected by proximity to noise floor and switching noise of regulator)
- Software
  - Implement Neural Network and Fuzzy Logic Control natively on embedded device
    - Help with controlling control latencies
  - Debug utilization of MBED OS for real-time task scheduling and resource sharing
  - Further test and debug Fuzzy Logic code
  - Build simulation models of control approaches before testing on hardware
  - Achieve Wi-Fi or Bluetooth data transmission simultaneously with system operation, especially with dedicated data transmission hardware available.