King Fahd University of Petroleum and Minerals
College of Computer Sciences and Engineering
Systems Engineering Department

CISE 490: Senior Design Project
(Term 191)

Path-Defined Solar-Powered Robot Using Artificial Intelligence

Advisor: Dr. Abdulwahed Al-Saif
Co-Advisor: Dr. Mujahid Al-Dhaifallah

Bin Mohaya, Turki 201569090
Omar Majeed 201523450

Abstract

This project is an attempt to build a self-managed robot in which it can generate power for itself, use soft computing (mainly Artificial Intelligence) and, thus, replace labor work jobs in the next decades. This report covers the hardware part of the robot, the software part, further analysis using Neural Networks, and industrial applications.

December 7, 2019

ACKNOWLEDGEMENT

At this stage, we would like to express our gratitude and sincere thanks to all our university professors who aided us during this project.

Thank KFUPM, SE department, Dr. Abdulwahed AL-Saif, and Dr. Mujahid Al-Dhaifallah. We would like, also, to express sincere thanks to all our colleges at KFUPM

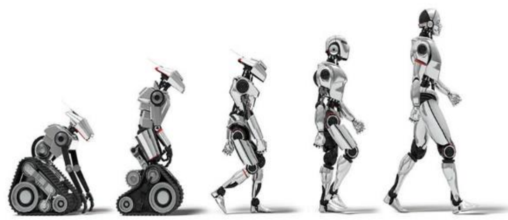
TABLE OF CONTENTS

LIST OF TABLES AND FIGURES

TABLES

FIGURES

CHAPTER 1



Source: robohub.org

1.1 INTRODUCTION

The world, nowadays, is rapidly changing into more industrial automation and self-dependent robotics. Scientists are devoting time and effort to efficiently and effectively link theory and practice. Heavy research and investigation have taken place in the field of Artificial Intelligence (AI), where machine learning, deep learning, and neuro-fuzzy systems are expected to be leading the next automation century. In this senior design project, we present an attempt to build a self-managed robot by applying concepts of AI as well as Green Energy. Initially, we state the objective of the project and what is supposed to be done. After that, we cover the hardware section of the report where we describe components and configuration in addition to circuit diagrams. The next stage is related to the software part of the project in which we explain the process and the automation concepts behind it. An analysis phase is conducted, next, on the gathered data. Finally, we discuss future advances and applications.

1.2 OBJECTIVE

In this project, we aim to build and assemble an intelligent robot that can handle itself in terms of power and control to substitute labor men work. To accomplish that, we introduced the simple function path-defined follower robot function in which the machine can decide where to move based on sensing a specific path in its environment (Usually considered to be a colored line laid on the ground).

CHAPTER 2

2.1 HARDWARE

In this project, several hardware elements were assembled to achieve the desired goals (Intelligence and Independency). Since we introduced the function of line tracking as the testing function, we used ultrasonic short-range sensors that can detect ground lines in terms of voltage change. The microcontroller is the raspberry pi 3b+ which uses Linux as its main operating system and python programming language for software development. The signal conditioning element is the Sabretooth DC motor driver and, thus, two motor drivers were used to move the robot with a passive wheel for balancing the turning. Lastly, Female-female and Male-male wires are used to connect nodes between elements.

2.1.1 Tetrix Parts

We used a packaged of hardware parts that are the Tatrix. TETRIX is a robotics building system which consists of aluminum elements for construction and powerful drive motors. These components are complete hardware robots with a number of different designs that are depend on the function of the robot itself.

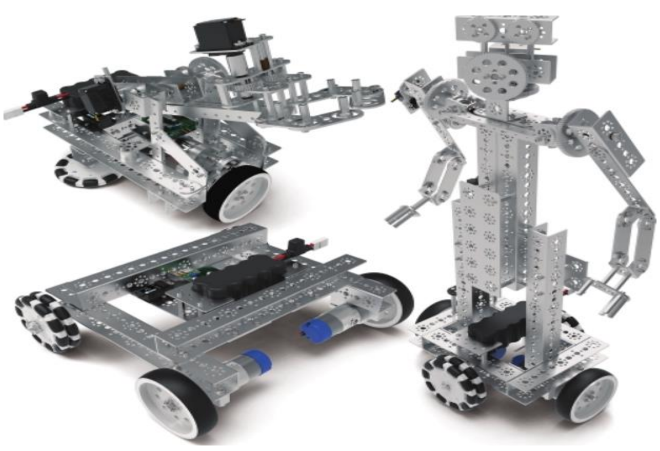


Figure 1: Tetrix Parts with different designs

Figure 2: TETRIX MAX Rechargeable 3,000 mAh

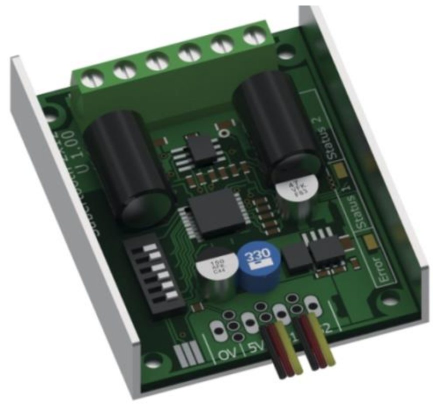


Figure 3: TETRIX MAX R/C Motor Controller

2.1.2 Battery

Here we used a TETRIX MAX Rechargeable 3,000 mAh, 12V NiMH Battery Pack. NiMH cells are good for several hundred cycles if properly charged and maintained. Battery needs to be fully charged before use. It should be charged and discharged a minimum of five times to attain full charge capacity see figure 2.

2.1.3 Sabretooth RC 2x12

Here we used TETRIX MAX R/C Motor Controller. This component can control two Dc motors and it connects them to the battery. It resembles here the signal conditioning element. See figure 3.

2.1.4 DC Motor

TETRIX MAX DC Motor has been used which has these specifications

1. Max Operating Power: 12V, 1.2A
2. Load Torque: 3.9 kg.cm
3. Load Speed: 137.5 rpm
4. Size: 100 mm x 37 mm
5. Weight: 222.8 g per motor
6. Shaft: 6 mm D-shaft

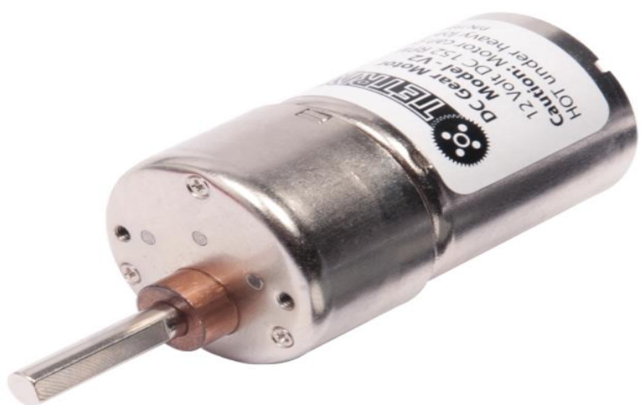


Figure 4: Tetrix 0.6 A DC Motor

2.1.5 Raspberry Pi 3B+

We used The Raspberry Pi 3B+ as the microcontroller. It is the latest product in the Raspberry Pi 3 range, boasting a 64-bit quad core processor running at 1.4GHz, dual-band 2.4GHz. Also, it allows wireless communication with 5GHz, LAN communications, and Bluetooth. However, to the sake of this project all wireless communication will not be adopted.

Specifications of the Raspberry Pi 3B+:

SOC: Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC
CPU: 1.4GHz 64-bit quad-core ARM Cortex-A53 CPU
RAM: 1GB SDRAM
WIFI: Dual-band 802.11ac wireless LAN (2.4GHz and 5GHz) and Bluetooth 4.2
Ethernet: Gigabit Ethernet over USB 2.0 (max 300 Mbps). Power-over-Ethernet support (with separate PoE HAT). Improved PXE network and USB mass-storage booting.
Video: Yes – Video Core IV 3D. Full-size HDMI
Audio: Yes
USB 2.0: 4 ports
GPIO: 40-pin
Power: 5V/2.0A DC power input
Operating system support: Linux and UNIX

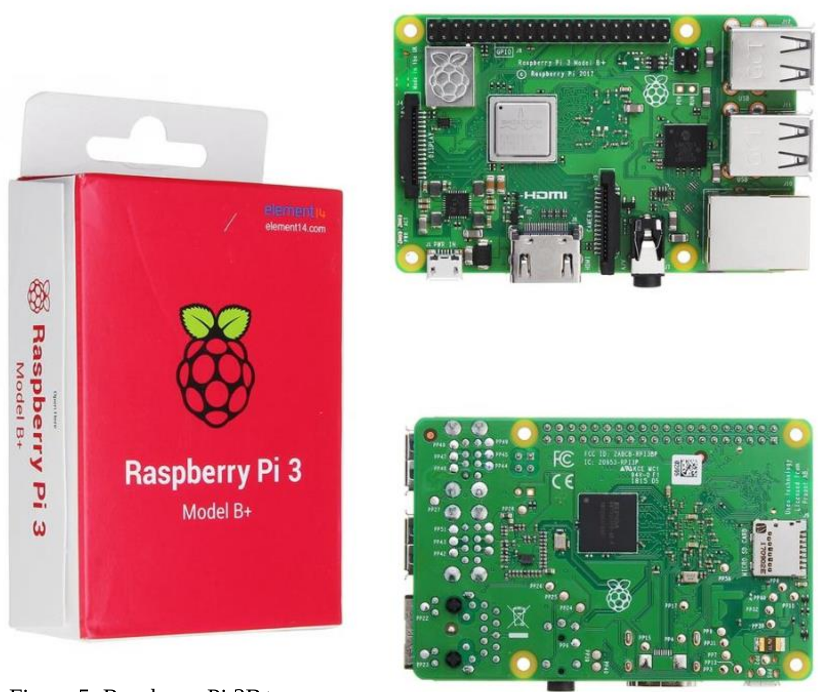


Figure 5: Raspberry Pi 3B+

2.1.6 Solar Panel

We used a solar panel with 18V and 20W rated power with the following specifications.

Max Power: 20W
Max Power Voltage: 18.5V
Max Power Current: 1.11A
Open Circle Voltage: 22.14V
Net Weight: 1.6KG
Product size: 410 x 340 x 18mm
Frame Material: Aluminum.
Output Tolerance: 5%.

Figure 6: solar panel with 18V 20W

2.1.7 Solar Charge Controller

We used a device that has capability to control the charges that come from the solar panel and this device is called Solar Charger Controller.

Solar Charger Controller specifications:

Rated Discharge Current: 20A
USB Output Voltage: 5V/3A
Battery Voltage: 12V/24V

Figure 7: Solar Charge Controller

2.1.8 Tracking Sensors

TCRT5000 infrared reflective sensor see figure 8.
Detects the reflected distance: 1mm ~ 25mm Applicable
The working voltage: 5V
A fixed bolt hole for easy installation
Small board size: 3.5cm x 1cm
A single weight: 4.5g

Wiring:

VCC: positive power supply (5V)
GND: negative one
OUT: Output

Figure 8: TCRT5000 infrared reflective sensor

2.2 VARIABLES (Inputs, Outputs)

This project resembles a Multi Input Multi Output Complex system (MIMO). There are four inputs to the system corresponding to two outputs. Three inputs are generated from the three infrared path detecting sensors attached to the bottom of the robot. The fourth input comes from the solar panel representing the main source of power to the system. These inputs are used to generate the two output signals to the two motors; each signal can independently control the speed of a motor. By adjusting different speeds, the controller can change the direction of motion, hence, it works as a mechanical differential element. In the next section, we provide the complete control block diagram.

2.3 CONTROL BLOCK DIAGRAM

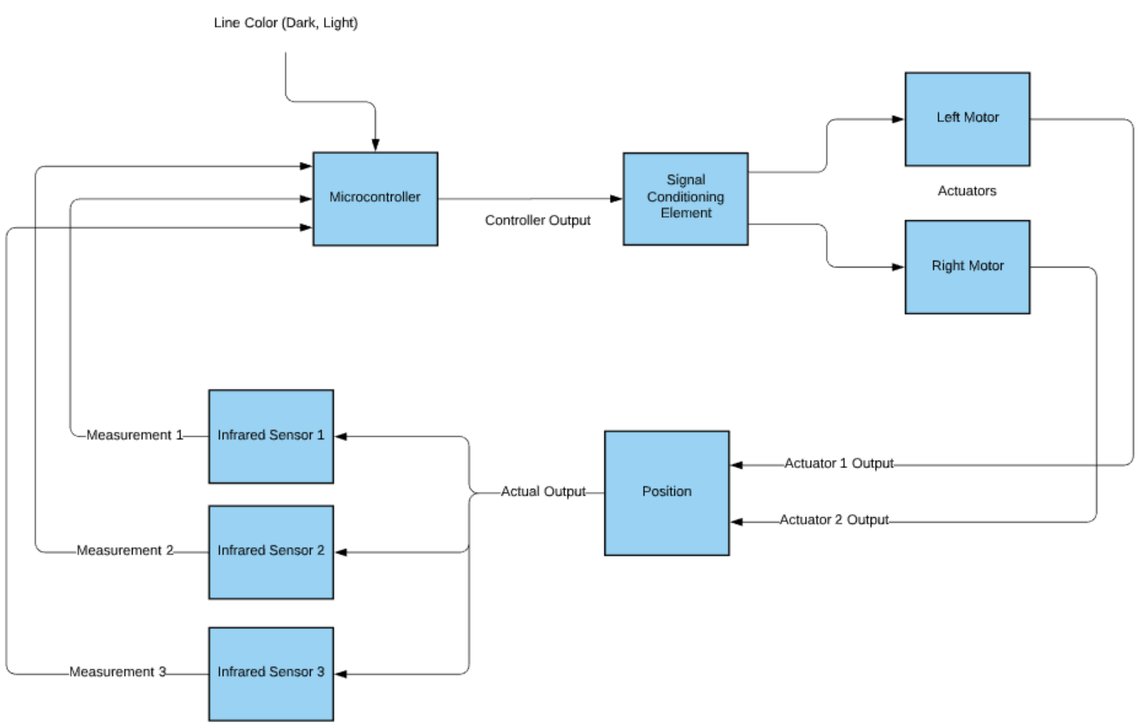


Figure 9: Control Block Diagram of The Robot

2.4 CIRCUIT DIAGRAM

To visualize how all parts are connected and function together, we provide the following circuit diagram drawn using fritzing software.

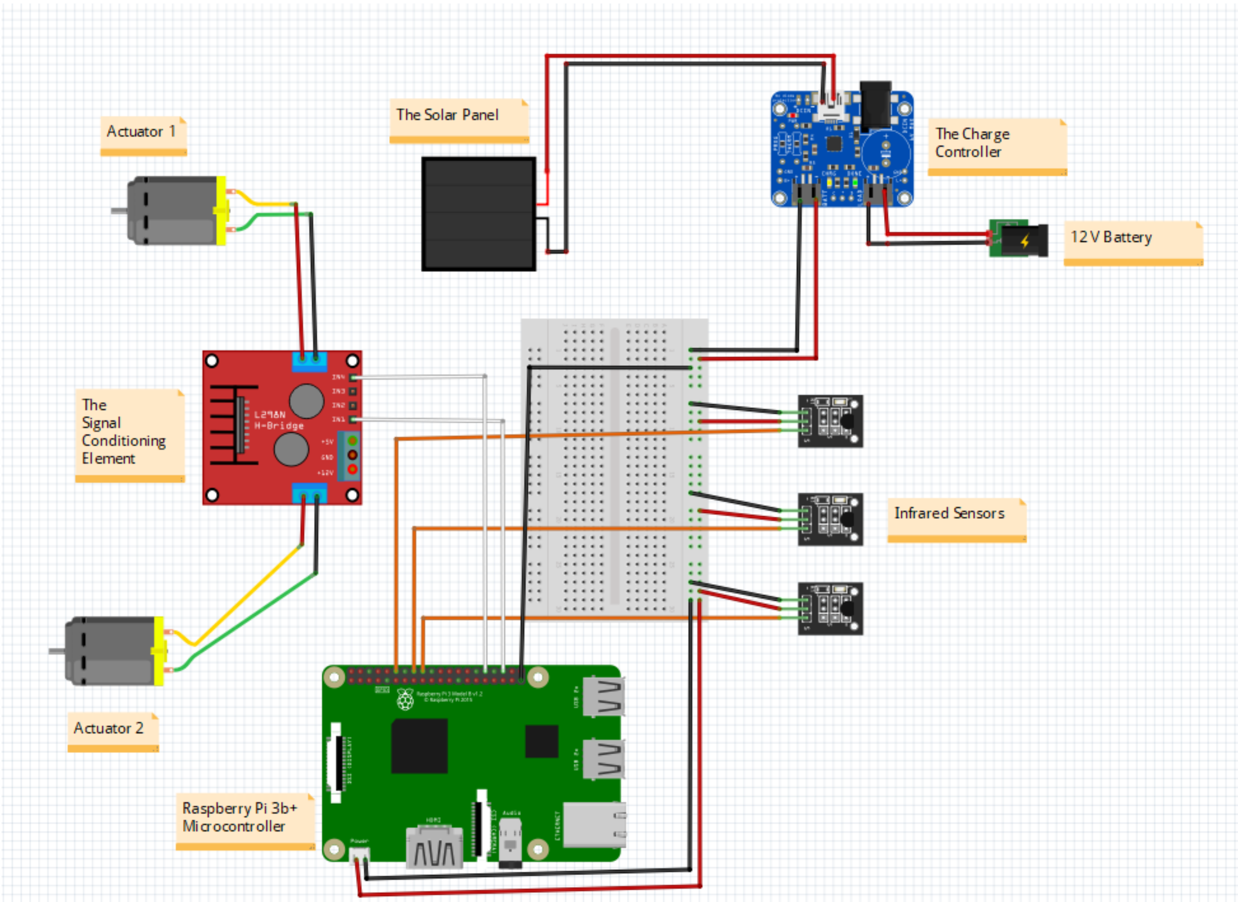


Figure 10: Complete Circuit Diagram

2.5 FINAL ASSEMBLY

All the above described hardware components were assembled to establish the desires objective. To summarize the hardware part, we can generally say that the system gets power from the solar panel; the charge controller uses optimization algorithms to provide maximum power to charge the battery. When the battery is fully charged the charge controller start discharging it. As the process of discharging occur, the microcontroller and all other elements start functioning. The following picture illustrates the final assembly of the robot.

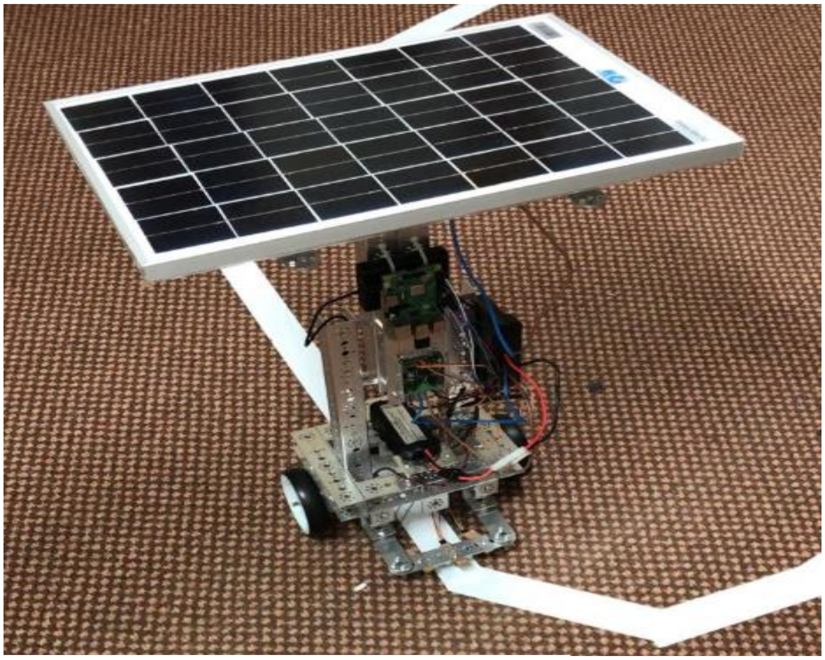


Figure 11: The Robot Final Assembly

2.6 POWER

The following table illustrates the power dissipated in each element of the system, thus, provide a measure for all power consumed.

ITEM	Operating Voltage (V)	Operating Current (A)	Operating Power (W)	Quantity	Total Power (W)
Infrared Sensor	5	0.02	0.1	3	0.3
DC Motor	12	0.6	7.2	2	14.4
DC Driver	5	1	5	1	5
Solar Charger	12	0.01	0.12	1	0.12
Microcontroller	5	2.5	12.5	1	12.5
				Total =	**32.32**

Table 1: Items Power Specifications

The battery used is able to generate 3A current in the discharge mode. If we calculate the generated power, we would arrive at the following

$$P = IV = 3(12) = 36\,Watts$$

We can see, from above formula, that the battery can efficiently handle the load (32.32 W). In the following table we present data gathered at discharging time.

Open Circuit Voltage (V)	Remaining Percentage Life
13.00	100 %
12.90	90 %
12.81	80 %
12.70	70 %
12.60	60 %
12.51	50 %
12.40	40 %
12.30	30 %
12.21	20 %

Table 2: Battery Discharge Voltages

The charging controller is programmed to disconnect the battery at an Open Circuit Voltage value of 13.2V (40%) for longer battery life. We have observed that the battery needs 1.5-2 hours to reach this voltage from being fully charged. Since the solar panel can provide a maximum of 1.11A with 75% efficiency, the following formula calculates the charging time

$$t = \frac{I(Ah)}{eA} = \left(\frac{3Ah}{0.75(1.11)}\right) = \ 4\,hours$$

Taking into consideration shadings and different light angles, we observed that the time for fully charge from a state of 40% is approximately 4 – 6 hours.

CHAPTER 3

3.1 MAXIMUM POWER POINT TRACKING THEORY

Today, governments and scientist around the globe dedicate time and effort to develop methods and products that extract the maximum available power from renewable sources (e.g. solar power and wind energy). The main motivator to that is the reality that fossil fuel energy is vanishing. In this chapter we introduce the theory behind the concept of Maximum Power Point Tracking for Photovoltaic Systems (MPPT) in a brief manner.

To understand MPPT, one must first have a cursory glance at the general characteristics of a Photovoltaic System. This is achieved by inspecting the Voltage – Current plot. The following figure illustrates generally that relationship.

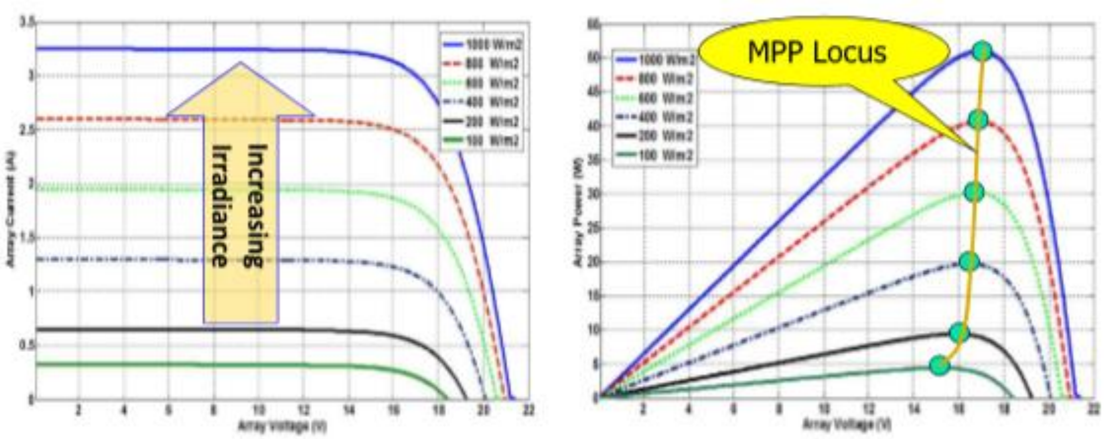


Figure 12: PV Characteristics I

We can note, from the above relationship graph, that the short circuit current drops significantly at higher open circuit voltage values. Thus, the maximum power is gained at the knee of the I-V curve. Also, we note that more irradiations cause an increase in the open circuit voltage as well as the short circuit current. Thus, the delivered power increases.

To see the complete picture, we now present, in the following figure, the effect of varying temperature values on the demonstrated characteristics above.

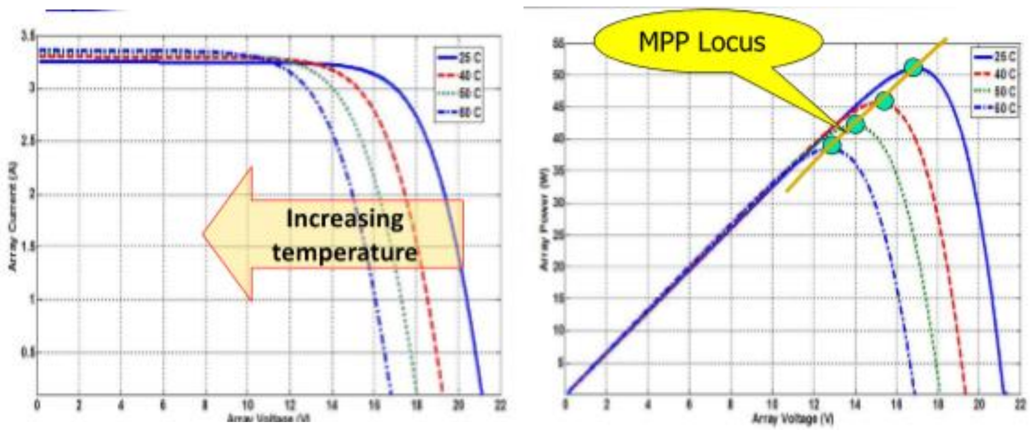


Figure 13: PV Characteristics II

As the above figure illustrates, increasing the temperature negatively affects the delivered power; it reduces the open circuit voltage, while slightly increasing the short circuit current.

Hence, the concept of MPPT takes priority in terms of detecting the values of voltage and current that delivers the maximum available power.

3.2 GENERAL MPPT METHODS

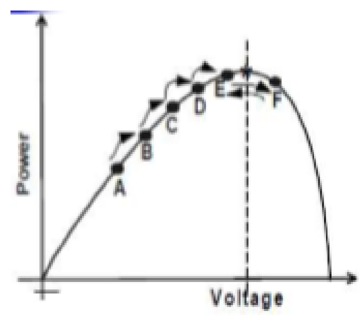
Now that we have demonstrated the characteristics of PV systems and the importance of the concept of MPPT, we next present three main algorithms that solve the problem (find the maximum power point irrespective of load and other conditions).

3.2.1 Fixed Voltage (Or Current) Method

This method assumes that the ratio of the open circuit voltage (short circuit current) to the V_{mpp} (to the I_{mpp}) is constant. Usually assumed to equal some ratio k varying from 70 to 85%. the following equation translates the concept of this method.

$$V_{mpp} = kV_{op}$$

3.2.2 Perturb and Observe Method



$$V_{ref}(k) = V_{ref}(k-1) + \alpha \cdot sign\left(\frac{\partial P}{\partial V}(k)\right)$$

$$sign(x) = \begin{cases} 1 & \text{if } x > 0 \\ -1 & \text{if } x < 0 \end{cases}$$

Figure 14: Perturb and Observe MPPT Method

This method regulates the voltage value based on the assumption that P-V graph resembles a bill shape. For instance, if the change in power relative to voltage is positive (increasing slope) the operating voltage increases; if it is negative, the operating voltage decreases until an equilibrium point is met. This algorithm is demonstrated in the following diagram:

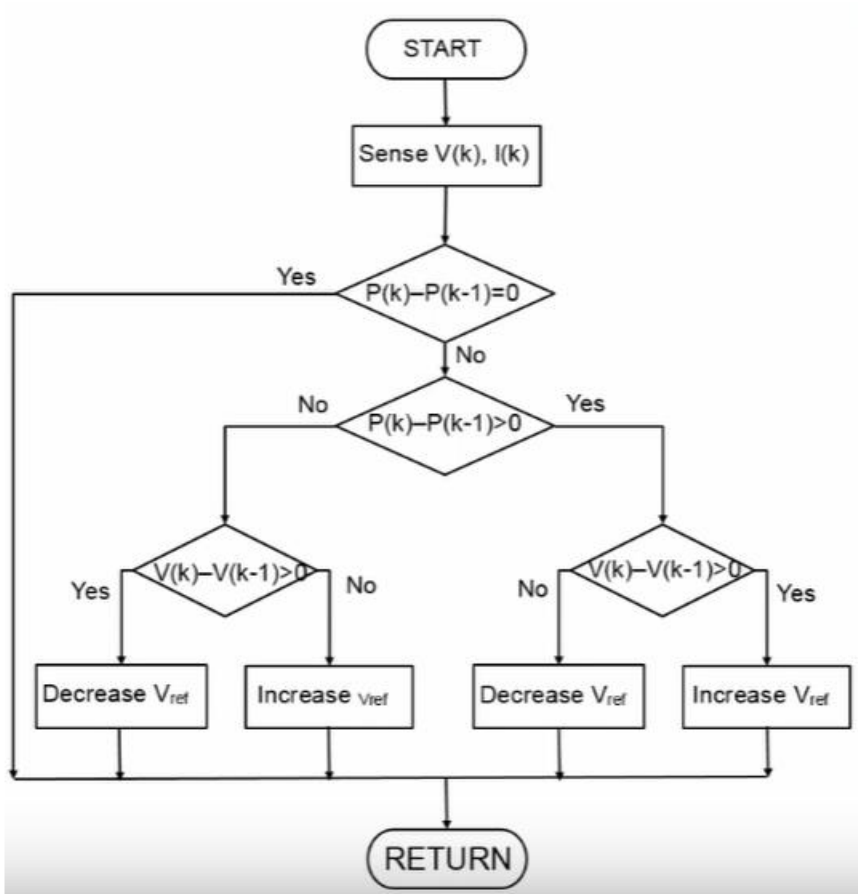


Figure 15: Perturb and Observe MPPT Method Algorithm Diagram

3.2.3 Incremental Conduction Method

This is the most robust and efficient practical method for detecting the maximum power and it is, hence, utilized in this project. It is based on the concept that the maximum power occurs when the instantaneous power rate of change with respect to voltage is exactly equal to the negative instantaneous conductance. This method can be visually illustrated in the following figure:

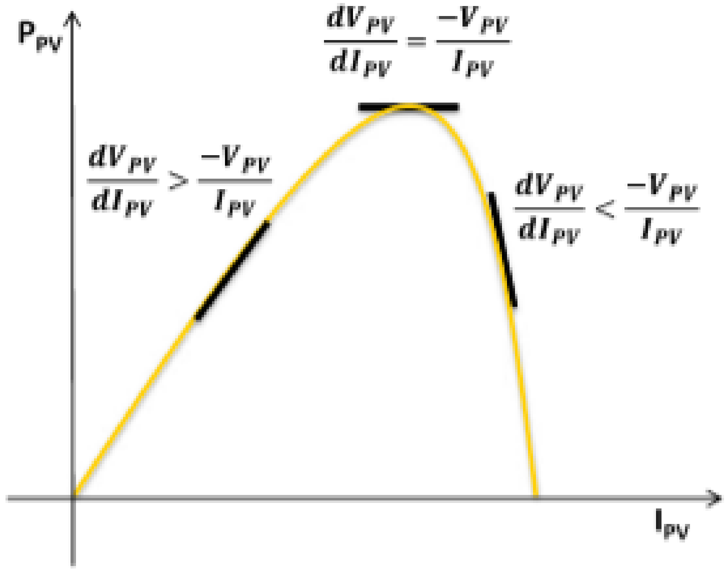


Figure 16: Incremental Conduction MPPT Method

As shown above, the controller compares the rate of change values to the conductance value iteratively to reach out the maximum point. To implement such algorithm, we provide the following diagram:

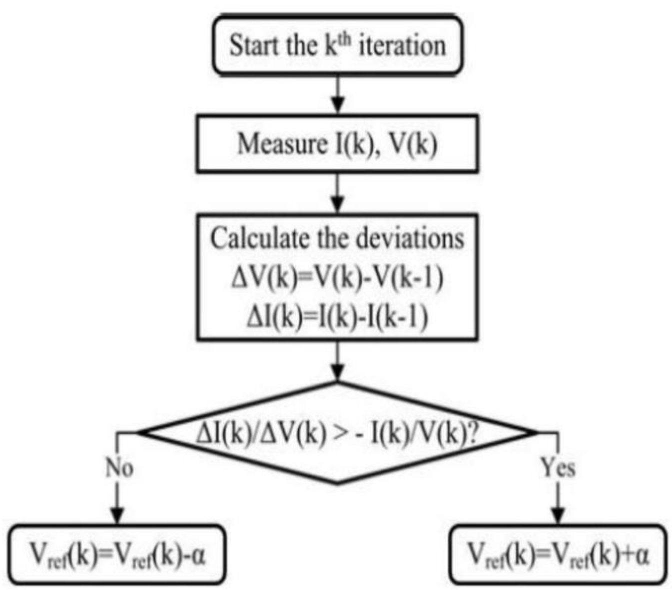


Figure 17: Incremental Conduction MPPT Method Algorithm

To utilize the maximum power provided by a Photovoltaic System, different methods are implemented on microcontrollers such as the Fixed Voltage Method, Perturb and Observe Method, and the Incremental Conduction Method. In this project, we utilize the third method, that is the Incremental Conduction.

CHAPTER 4

4.1 SOFTWARE ALGORITHMS

In this chapter we cover the software algorithm part for operating the system. There are two main ways to installed algorithms. First, a fuzzy approach is used in designing the robot controller. Second, to maximize the generated power we apply an optimization algorithm through the charging controller.

4.1.1 The Fuzzy Controller

Since the system represent a complex MIMO mechanical system, we choose an artificial intelligent algorithm (mainly Mamdani Fuzzy Model) to smoothly control its movement without figuring out the complete mathematical model behind it. Initially we define the inputs as the three signals generated from the infrared sensors; the outputs are the two motors' signals. The tool used to design the fuzzy controller is MATLAB Fuzzy Toolbox. The following figure illustrates the general structure of the fuzzy model.

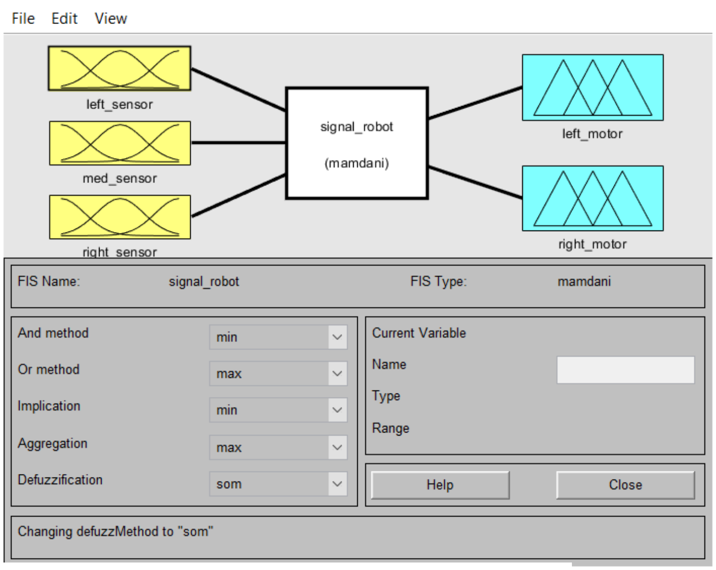


Figure 18: Fuzzy Controller General Structure

Two membership functions were used for each input (HIGH or LOW). And three member functions for the outputs (Fast, Med, and Slow). The following figure shows the ranges and membership functions for all inputs and outputs.

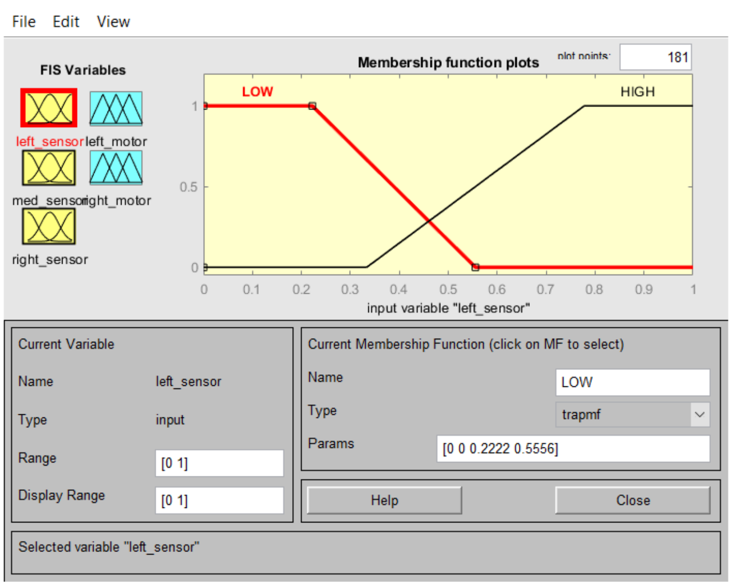


Figure 19: Input Membership Functions

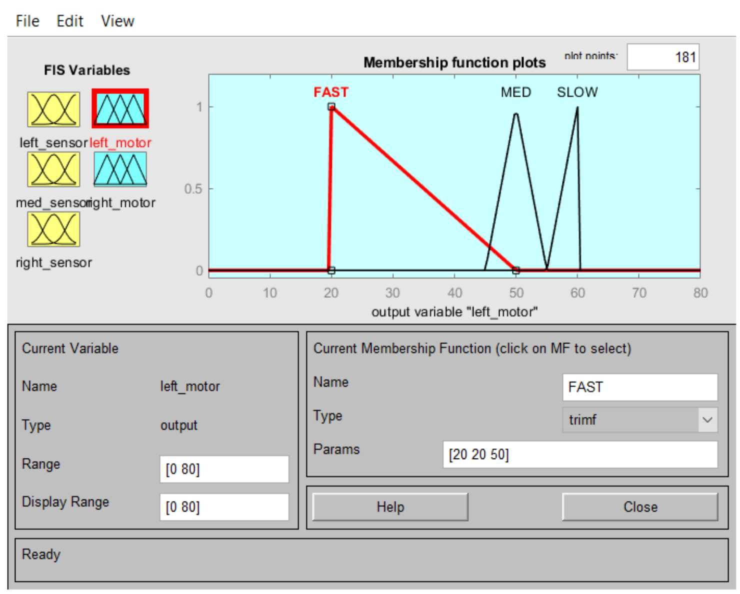


Figure 20: Output Membership Functions

Five rules were implemented in the system such that each rule can provide a direction of motion to the robot. The following table explains the designed rules.

Left Sensor	Meddle Sensor	Right Sensor	Left Motor	Right Motor
LOW	LOW	LOW	SLOW	SLOW
HIGH	HIGH	HIGH	SLOW	SLOW
HIGH	LOW	HIGH	MED	MED
LOW	HIGH	HIGH	SLOW	FAST
HIGH	HIGH	LOW	FAST	SLOW

Table 3: Controller Rules

4.1.2 Power Optimization and Safety Algorithms

To ensure maximum power while charging, the charge controller applies algorithm that do the following

- Ensure enough current with minimal shading areas, this is accomplished by obtaining a higher capacity panel along with the charge controller
- Prevent short circuits and overload
- Prevent reverse direction of current
- Provide regular measurements of power consumed
- Cut the load at 40 % remaining battery life
- Measure environment temperature to cut load for 20 min if over heated
- Apply Incremental Conductance MPPT Method

CHAPTER 5

5.1 STATISTICAL ANALYSIS

In this chapter we cover the part of analyzing the project data. Using Pandas Data Frame, we were able to generate 910 samples from the robot while in operating mode around a circular path. We present, hence, different steps in analyzing the data for more insights and understanding. To analyze the data, we used Anaconda Jupyter Notebook along with the MATLAB Neural Network Toolbox. The following figure shows a sample table from the data.

In [1]:
```
import pandas as pd
```

In [21]:
```
df = pd.read_csv('signal_robot_data.csv')
df.sample(5)
```

Out[21]:

	left_sensor	med_sensor	right_sensor	sampling_time	left_motor	right_motor
417	1	0	1	0.000451	1.782	1.782
51	1	1	1	0.000298	1.782	1.782
737	1	1	1	0.000303	1.936	1.694
247	1	1	1	0.000412	1.694	1.936
326	1	0	0	0.000347	1.694	1.936

Table 4: Sample Data

There are seven columns in the data set; the first column represents the indexes. left_sensor, right_sensor, and med_sensor represent the three sensor measurements in High or Low voltage. The sampling time represent the time taken by the machine to generate an output in responding to an input in seconds. The last two columns represent the output voltage from the microcontroller in Volts ranging from 0 – 3.3V. This output is next fed to the signal conditioning element.

To first understand the system, let us have a quick lock on general statistical estimators. The following table does that.

	left_sensor	med_sensor	right_sensor	sampling_time	left_motor	right_motor
count	910.000000	910.000000	910.000000	910.000000	910.000000	910.000000
mean	0.873626	0.778022	0.798901	0.000461	1.786642	1.831319
std	0.332453	0.415805	0.401042	0.000210	0.105892	0.108405
min	0.000000	0.000000	0.000000	0.000290	1.694000	1.694000
25%	1.000000	1.000000	1.000000	0.000319	1.694000	1.694000
50%	1.000000	1.000000	1.000000	0.000363	1.782000	1.782000
75%	1.000000	1.000000	1.000000	0.000496	1.936000	1.936000
max	1.000000	1.000000	1.000000	0.001522	1.936000	1.936000

Table 5: General Statistics for Each Variable

From the above table, we can generally say that the sampling time mean is approximately 0.46 ms, which is quite large relatively. The right motor average voltage is higher; this may be the case because of the circular path the robot was detecting. There is a high Standard Deviation in the sampling time. In the following figure we examine the variables distributions.

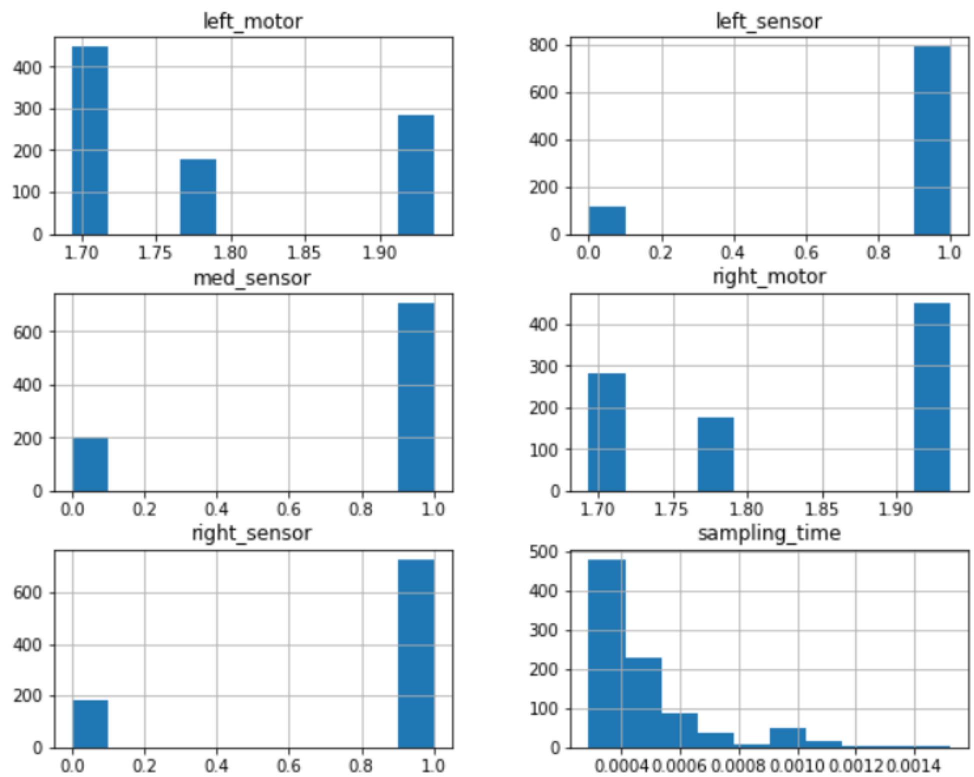


Figure 21: Variables Distributions

From above distributions, we can generally say that the controller is able to stabilize the robot looking at the two motors distributions. Moreover, the sampling time distribution is skewed to the right instead of being normally distributed. This fact may be due to the efficiency of the controller. We can, thus, conclude that sampling time is high which affect algorithm efficiency.

5.2 NEURAL NETWORK ANALYSIS

In this part of the analysis we utilize the MATLAB Neural Network toolbox to build and test a model for the robot. The type of analysis is the fitting supervised learning app. The training set constitutes 70% of the data, whereas each of the validation set and the testing set resembles 15% of the data volume. Moreover, one hidden layer with five neurons are applied to generate the robot model. The NN is trained 10 times before collecting the results. The following table shows the MSE result.

Results

	Samples	MSE	R
Training:	636	7.99688e-3	5.74563e-1
Validation:	137	8.48601e-3	5.27795e-1
Testing:	137	8.01436e-3	5.94408e-1

Table 6: The Neural Network Model Results

We can note that the MSE is very tiny which indicate that the model may be used for a better performance on the robot. Up to this point we did not apply the model and analyzed the operation using a neural network based controller.

In the next figure we express the performance graph of the model.

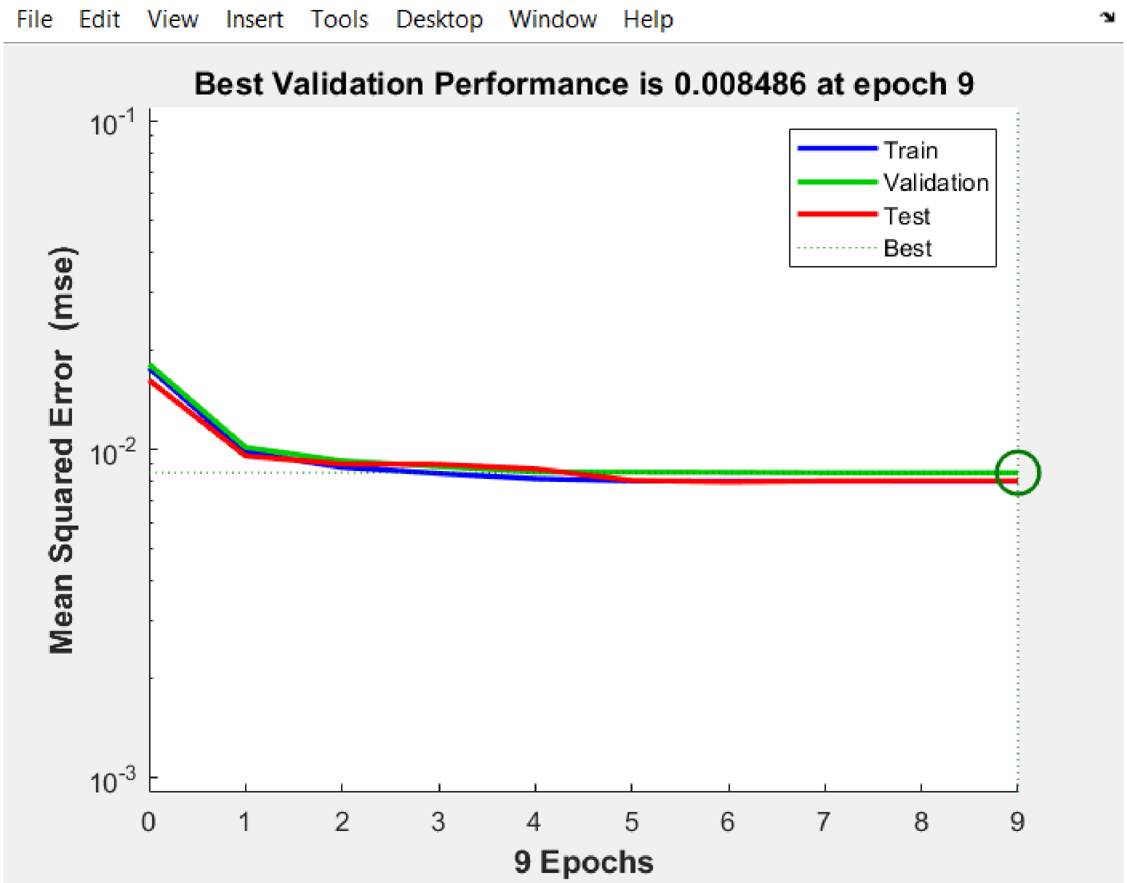


Figure 22: The Neural Network Model Performance

An astonishing performance is illustrated by the one-hidden layer NN with five neurons. It required only 9 epochs to generate the model, thus, the system did not acquire any further increase in accuracy after that number of epochs.

To understand our data deeply, we present the error distribution in the next figure

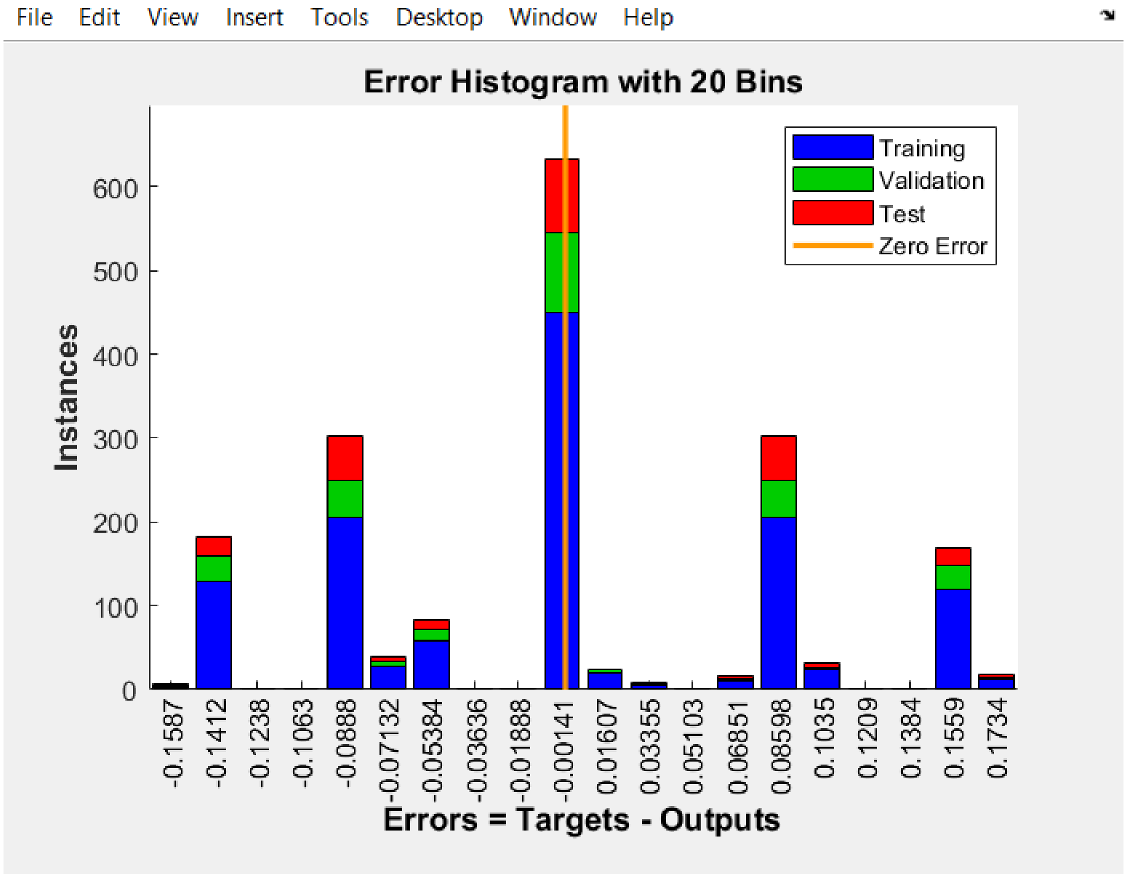


Figure 23: The Neural Network Model Error Distribution

The error distribution also indicated a better performance by the generated model. In future steps we will present another statistical analysis of the system using the neural network model.

CHAPTER 6

6.1 APPLICATIONS

Figure 24: Guidance System

Figure 25: Healthcare Robot

6.1.1 KFUPM Autonomous Delivery

We aim to use this robot as KFUPM deliver robot by generate a specific path and add a small box to carry goods.

6.1.2 Guidance System for Industrial Robots Moving on Shop Floor

A guidance system is a virtual or physical device, or a group of devices implementing a guidance process used for controlling the movement of a ship, aircraft, missile, rocket, satellite, or any other moving object. See Figure 24.

6.1.3 Healthcare Applications

A medical robot is a robot used in the medical sciences. They include surgical robots. These are in most tele manipulators, which use the surgeon's activators on one side to control the "effector" on the other side. See Figure 25.

Figure 26: Warehouse Robots

Figure 27: Home Robot to Carry Food Dishes

6.1.4 Industrial Applications

Another common type of robot used in industry is those that move items around a factory floor or warehouse. There are giant robotic crane systems capable of shifting pallets in storage complexes. They receive instructions on where goods need to be moved from and to within shelving systems see Figure 26.

6.1.5 Service Applications

Automated restaurant or robotic restaurant is a restaurant that uses robots to do tasks such as delivering food and drinks to the tables and/or to cook the food. Such a robots are hired to replace human labor (such as waiters and chefs) see figure 27.

CHAPTER 7

7.1 FUTURE STEPS

Our next main objective is to implement advanced algorithms to generate the maximum possible power using the solar panels. Several algorithms have been invented; however, we aim to implement and test the Maximum Power Point Tracking Controller. In this algorithm the panel detects the maximum point that allows maximum power. Furthermore, we aim to introduce new objectives and functions for the robot to substitute labor men jobs (e.g. garbage collecting, floor cleaning, or carrying robot)

7.2 HARDWARE CONSTRAINS

Although all hardware parts have been assembled carefully and thoroughly, we present the following constrains:

- High outdoors temperature could result in failure of the sensors
- The passive wheel could stick in inclined areas and the robot no longer move
- Direct sunlight on the controller reduces its efficiency by increasing its temperature
- After a few hours of operation, the sensors require recalibration
- The charger USB port may suddenly cut the current, thus a safety power bank is used with the raspberry pi to prevent such cases

- The microcontroller (Raspberry Pi) has both python 2.7 and python 3.7 installed which may cause confusion when creating virtual environments
- The motors sound when operating is noisy

7.3 CONCLUSION

Since governments and managers around the globe hardly work to increase people knowledge and life quality, humans in the next decades will no longer accept labor work jobs. Therefore, Self-Powered robots must be introduced to modern life as this project shows. Linking Artificial Intelligence to robotics can provide a variety of functions to element labor jobs.

REFERENCES

- Pitsco Education Website < www.pitsco.com >

- Adafruit Website < www.adafruit.com >

- MATLAB Neural Network Toolbox
< https://www.mathworks.com/help/deeplearning/ref/nnstart.html>

- Pandas Documentation < www.pandas.pydata.org/pandas-docs >

- Raspberry Pi Website < www.raspberrypi.org >