Optimization of Irrigation System through IoT Methods & Machine Learning

Prem Rana

Southern Illinois University Carbondale

Introduction

The goal of this project is to create a fully autonomous irrigation system that reduces water usage and improves the quality of soil through a needs-based system.

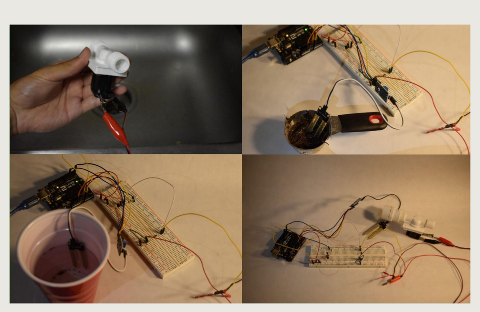
Built on the College of Agriculture Green Roof, this project aims to be a low-maintenance, low-cost, and scalable system that improves plant research conditions on a specific plot of land.

The system is built on a microcontroller "Queen Bee" system, where 3 different Arduino Microcontrollers collect data and send it to a "Queen" Raspberry Pi to interpret and store the data.

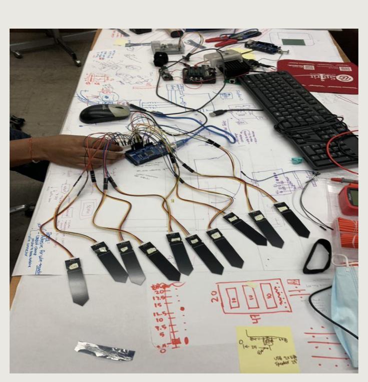
The system is built on a microcontroller "Queen Bee" system, where 3 different Arduino Microcontrollers collect data and send it to a "Queen" Raspberry Pi to interpret and store the data.

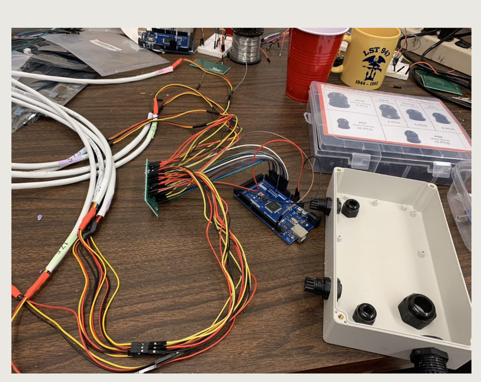
Methodology

This project was done in 3 phases. Phase 1 involved creating the timer-based system with a prototype solenoid valve



Phase 2 involved installing the "Queen Bee" system and collecting data from the soil medium. This also involved calibrating sensors and developing persistent arduino code.



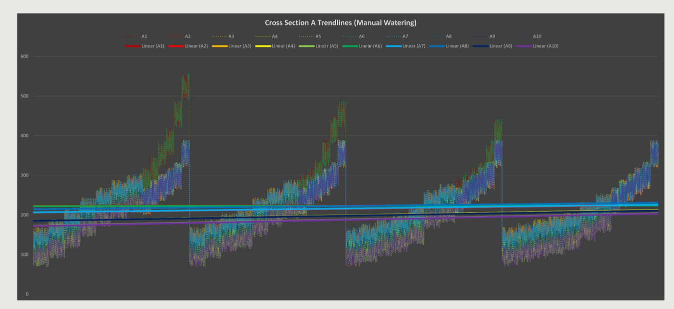


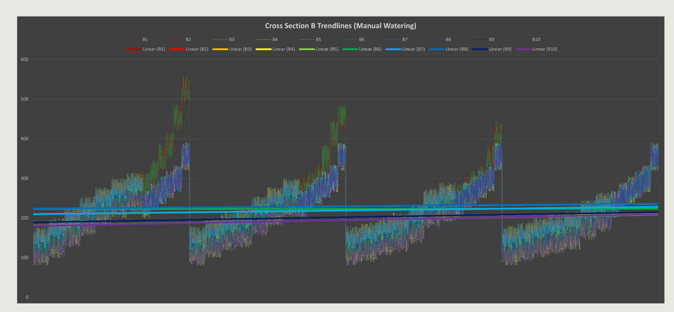
Phase 3 involved implementing a decision tree algorithm to predict soil moisture levels and the time for the soil to dry in order to optimize water usage.

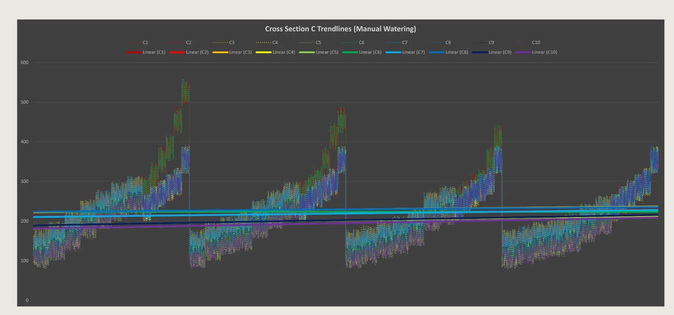
```
DecisionTreeRegressor(ccp_alpha=0.0, criterion='mse', max_depth=None,
                      max_features=None, max_leaf_nodes=None,
                      min_impurity_decrease=0.0, min_impurity_split=None,
                      min_samples_leaf=1, min_samples_split=2,
                      min_weight_fraction_leaf=0.0, presort='deprecated',
                      random_state=0, splitter='best')
```

Results

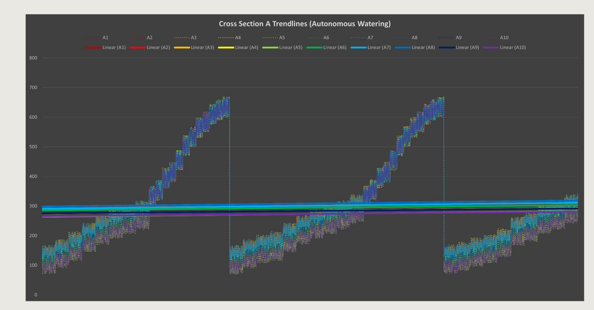
Manual Watering

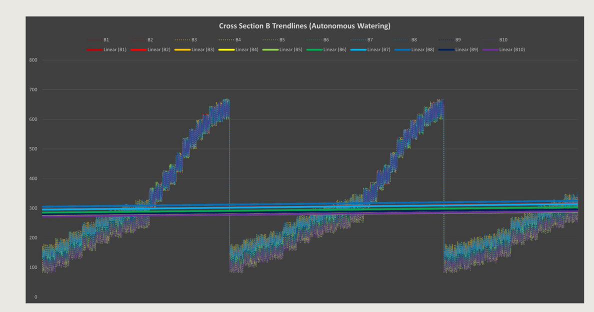


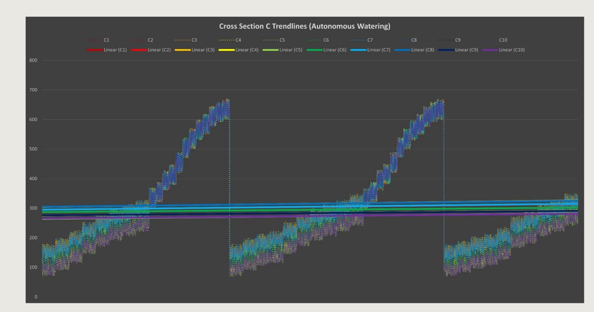




Autonomous Watering







The data above shows the recorded soil moisture over time, along with a general trendline of the data. There were two points collected, data from when the irrigation system was run manually (watered every day for 15 minutes) and data when the Raspberry Pi chose to water. The sample size is around 1200 timestamps collected, with a range of 4 days (around 300 samples per day).

The manual watering used 4 waterings, each using about **108.75** gallons per watering (Dripline flow rate of 7.23 gallons-per-minute, or gph). This meant that the manual watering system used about **433.8** gallons. In this same length of time, the autonomous watering system only used 2 waterings, meaning that the amount of water used was **217.52** gallons. This is a **49.8%** decrease in total water usage.

Average soil moisture in total was read at about a level of **212**. Each watering would have the moisture level decrease slightly, with the average moisture level of each subsequent watering decreasing **4%** (228 to 216 to 207 to 198). In comparison to the autonomous irrigation system, the average soil moisture on a similar sample size was read at a level of **216**, with a decrease from 238 to 209 (**12%)** following another decrease from 209 to 203 (**2%)**. It's worth noting that the lower a reading is, the wetter the soil is. Here, the average moisture level is higher due to the soil being watered when it needs to, rather than at an arbitrary point in time.

For the machine learning model, a Decision Tree Regressor was chosen to predict how long the soil would take to dry. This model was specifically chosen due to it's ability to handle various features (such as temperature and weather). Data was averaged on a cross-sectional basis (how long it took on average for each cross section to dry). This model with no cross-fitting or changing of default values had a accuracy of **74%.**

Conclusion

In conclusion, it has been proven that an autonomous irrigation system that is low-cost, easily maintainable, and scalable, has the potential of saving nearly 50% of normal water usage. While still a preliminary model, there is lots of potential for a system like this to be scaled.

Some improvements include developing a more enclosed setup, as the wires were often exposed to the weather causing unreliability. Furthermore, experimenting with radio frequencies to communicate data instead of using wires may help with maintenance. This is due to there being many points of failure in the wired system (connections failing, short circuiting, etc) that cause the data to be incorrect or inaccurate. Ideally, this system would be able to run without internet connection and would simply be plug and play. However, the reality is that much of the data relies on the internet (specifically for weather), and there is many preliminary steps currently needed to setup the system. Future iterations may include a custom-designed sensor to better control the data collected

While 74% accuracy for a machine learning model is generally underwhelming, in this instance it is good. There were little modifications done to the actual model, so to have a decent accuracy by default is promising. Future iterations will revisit the Decision Tree Model, and compare to other machine learning models. More focus will also be placed on pruning the data and conducting feature-engineering to create a more useful dataset.

Acknowledgements

This project would have not been possible without the support of the following groups:

- Gary Kinsel and the Office of the Vice Chancellor of Research for funding this project
- Dr. Koushik Sinha for serving as the primary mentor
- Dr John Groninger for allowing research on the SIU Green Roof
- The Green Roof Team for the support and assistance towards development of this project