AI-VVO

Design Document

Sdmay21-24
Gelli Ravikumar
Abdul-Salam Andedoja
Ian Kegley
Jacob Gleason
Rene Chavez
Tyler Norris
Sdmay21-24@iastate.edu
https://sdmay21-24.sd.ece.iastate.edu
Revised: 10/04/2020/Version 1

Executive Summary

Development Standards & Practices Used

List all standard circuit, hardware, software practices used in this project. List all the Engineering standards that apply to this project that were considered.

Summary of Requirements

- Take input from sensor readings
- Use sensor readings to formulate and design Machine Learning / Deep learning algorithms
- Store sensor reading data on Google Cloud Platform
- Take desired output from Machine Learning / Deep learning algorithms to display information on our Main page
- Design a Webpage that displays various sensor data through scripting

Applicable Courses from Iowa State University Curriculum

- Com S 227
- Com S 228
- Com S 309
- Com S 319
- Com S 311
- Math 207
- Math 165/166

New Skills/Knowledge acquired that was not taught in courses

List all new skills/knowledge that your team acquired which was not part of your Iowa State curriculum in order to complete this project.

- Design and implement Machine Learning / Deep Learning algorithms
- Use Google Cloud Platform

Table of Contents

1 Introduction

1.1 Acknowledgement

Dr. Gelli Ravikumar

1.2 Problem and Project Statement

This project aims to design and implement cloud-based machine learning or deep learning algorithms for Volt-VAR control (VVC) and Volt-VAR optimization (VVO) for distributed energy resources integrated distribution grid.

1.3 Operational Environment

This project operate using Google Cloud Platform Environment

1.4 Requirements

- Collecting data streams and publish in the data pipelines

- Communications - HTTP / MQTT

- Design and implementation of ML/DL algorithms for VVC and VVO application

- Design and implementation of ML/DL algorithms for data analytics or anomalies over the real-time data streams

- Dashboard using client-side (front-end) scripting to visualize the data, plots, and analytics,

- Test and validate the application with available distribution grid simulators such as OpenDSS and GridAPPS-D

1.5 Intended Users and Uses

This project could be used by researchers to see how our particular algorithm responds to different situations in the distributed energy grid.

1.6 Assumptions and Limitations

Assumptions

- Our users will consist of one due to our client being our faculty advisor

Limitations

- This project will not get data directly from a real energy grid, it will instead receive simulated data in the form of a .json file.

- Budget consists of $300 Trail Credits to be used for our Google Cloud Platform workspace

1.7 Expected End Product and Deliverables

The first main deliverable of this project will be the machine learning algorithm our team designs to manage the energy grid. The algorithm must be able to be trained quickly.

The second deliverable will be a website designed by our group. This website will be receiving data that is output from our algorithm, displaying it in an easy to digest way. The website will consist of multiple graphs and figures that will assist the viewer.

2 Project Plan

2.1 Task Decomposition

- Collect data sets from OpenDSS Distribution Grid
    - Retrieve Data Sources (csv, json, etc)
- Establish direct access via MQTT/HTTP from the Distribution Grid to our Google Cloud Platform workspace
- Design and implement our ML/DL algorithm for VVC and VVO application
- Design and implement our ML/DL algorithm for data analytics or anomalies over the real-time data streams
    - Begin training our system in our environment
    - Begin testing our system in our environment
- Design Frontend Dashboard
    - Display datasets from our Distribution Energy Resources
    - Frontend scripting to visualize the data, plots and analytics
- Test and validate application from available Distribution Grid simulators

2.2 Risks And Risk Management/Mitigation

Possible risks in this project include;

- The possibility of our ML/DL algorithm won't be able to truly optimize our system at the rate that we want.

- Some functionalities of the application may not be doing their task correctly or efficiently

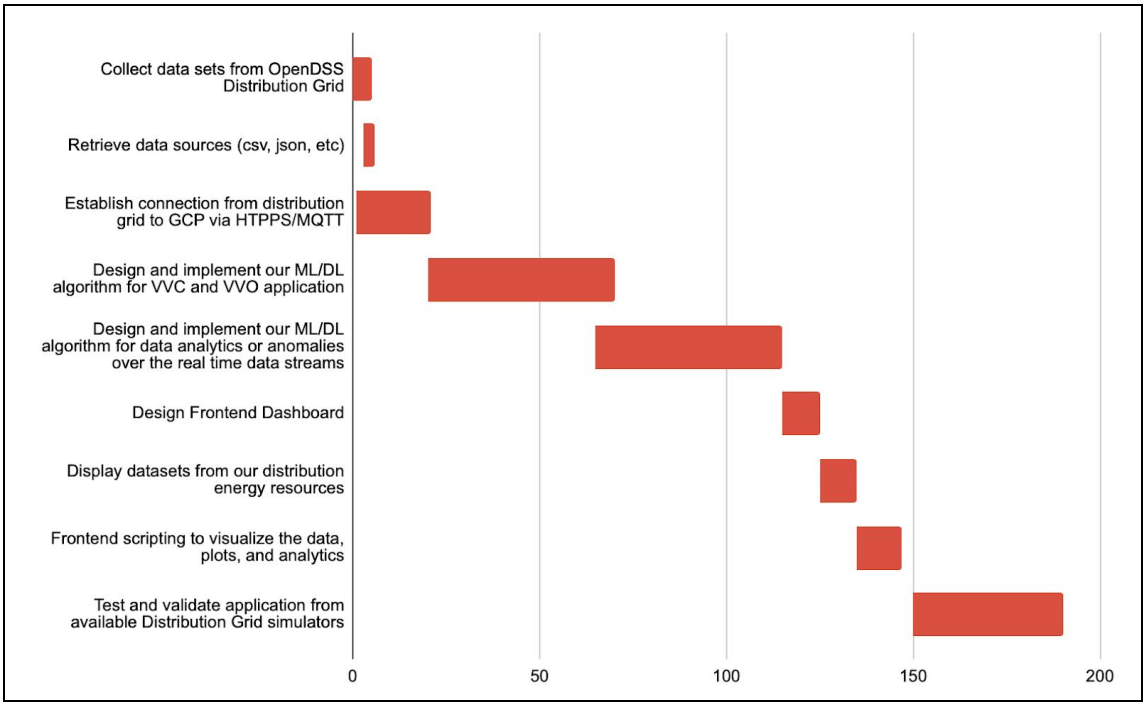
- Connection for our Google Cloud Platform workspace to our Frontend Dashboard could be lost

2.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

- *Collect data sets from OpenDSS Distribution Grid* - **Receive data sets, this data being collected will be from simulated energy resources.**
    - *Retrieve Data Sources (csv, json, etc)* - **Receive json file from faculty advisor**
- *Establish direct access via MQTT/HTTP from the Distribution Grid to our Google Cloud Platform workspace* - **Establish connection to our Google Cloud Platform with Django implementation**
- *Design and implement our ML/DL algorithm for Volt-VAR Control and Volt-VAR Optimization* - **Design algorithm to be able to optimize our results for our Distributed Energy Resources integrated distribution grid**
- *Design and implement our ML/DL algorithm for data analytics or anomalies over the real-time data streams* - **Design algorithm to be able to optimize our results for our Distributed Energy Resources integrated distribution grid**
    - *Begin training our system in our environment* - ***Train system to optimize our data streams to at least a 75% efficiency rate increase***
    - *Begin testing our system in our environment* - **By testing system we want to ensure that our algorithm is functioning correctly and delivered our expected result**
- *Design Frontend Dashboard*
    - *Display datasets from our Distribution Energy Resources* - **Our required data from our Distribution Energy Resource are displayed on our dashboard.**
    - *Frontend scripting to visualize the data, plots and analytics* - ***See desired data on front-end client side dashboard***
- *Test and validate application from available Distribution Grid simulators* - **Test application to ensure all functions are running smoothly and efficiently**

2.4 Project Timeline/Schedule

Gantt Chart



We will start the project by collecting the data from the OpenDSS Distribution Grid Simulator using HTTP/MQTT communication requests. Once our data sources are being brought in, we can start developing and creating the machine learning/deep learning (ML/DL) algorithms for the VVC and VVO application as well as for checking the data for anomalies. Next, we will create the frontend dashboard to allow for visualization of the data output from the algorithms. Finally, we will test and validate the application using available Distribution Grid Simulators.

2.5 Project Tracking Procedures

Our team will be using the kanban board available in Gitlab to keep track of tasks and subtasks.

2.6 Personnel Effort Requirements

Task	Explanation	Person-hours
Collect data sets from OpenDSS Distribution Grid	Simulation data must exported from OpenDSS and given to our Algorithms for analysis	5
Retrieve data sources (csv, json, etc)	Collect the data sets from OpenDSS in a source file of a certain type with the types being a csv file, json file, or other	3
Establish connection from distribution grid to GCP via HTPPS/MQTT	Automation aspect of previous task, will allow data from simulation to go directly to our project on Google Cloud Platform	20
Design and implement our ML/DL algorithm for VVC and VVO application	Create machine/deep learning algorithms to take the Distribution Grid data from a data source file to an output data file	50
Design and implement our ML/DL algorithm for data analytics or anomalies over the real time data streams	Create a machine/deep learning algorithm that will identify anomalies in our data stream	50
Design Frontend Dashboard	Design the dashboard that will use that data output from our algorithms to create readable graphics	10
Display datasets from our distribution energy resources	Display the datasets produced by our simulation on our front end dashboard	10
Frontend scripting to visualize the data, plots, and analytics	Use libraries available to have our front end create plots and analytics based on the data from our algorithms	12
Test and validate application from available Distribution Grid simulators	Simulate our project for real-life use	40

Total hours: 200 hours

2.7 Other Resource Requirements

Other resources aside from financial required to complete the project are the PowerCyber workbench and virtual machines.

2.8 Financial Requirements

$300 - Google Cloud Platform trial credits

3 Design

3.1 Previous Work And Literature

Include relevant background/literature review for the project

– If similar products exist in the market, describe what has already been done

– If you are following previous work, cite that and discuss the **advantages/shortcomings**

– Note that while you are not expected to "compete" with other existing products / research groups, you should be able to differentiate your project from what is available

Detail any similar products or research done on this topic previously. Please cite your sources and include them in your references. All figures must be captioned and referenced in your text.

3.2 Design Thinking

Detail any design thinking driven design "define" aspects that shape your design. Enumerate some of the other design choices that came up in your design thinking "ideate" phase.

3.3 Proposed Design

Include any/all possible methods of approach to solving the problem:

- Discuss what you have done so far – what have you tried/implemented/tested?
- Some discussion of how this design satisfies the **functional and non-functional requirements** of the project.
- If any **standards** are relevant to your project (e.g. IEEE standards, NIST standards) discuss the applicability of those standards here
- This design description should be in **sufficient detail** that another team of engineers can look through it and implement it.

3.4 Technology Considerations

Highlight the strengths, weakness, and trade-offs made in technology available.

Discuss possible solutions and design alternatives

3.5 Design Analysis

- Did your proposed design from 3.3 work? Why or why not?
- What are your observations, thoughts, and ideas to modify or iterate over the design?

3.6 Development Process

Discuss what development process you are following with a rationale for it – Waterfall, TDD, Agile. Note that this is not necessarily only for software projects. Development processes are applicable for all design projects.

3.7 Design Plan

Describe a design plan with respect to use-cases within the context of requirements, modules in your design (dependency/concurrency of modules through a module diagram, interfaces, architectural overview), module constraints tied to requirements.

4 Testing

Testing is an **extremely** important component of most projects, whether it involves a circuit, a process, or software.

1. Define the needed types of tests (unit testing for modules, integrity testing for interfaces, user-study or acceptance testing for functional and non-functional requirements).
2. Define/identify the individual items/units and interfaces to be tested.
3. Define, design, and develop the actual test cases.
4. Determine the anticipated test results for each test case
5. Perform the actual tests.
6. Evaluate the actual test results.
7. Make the necessary changes to the product being tested

8. Perform any necessary retesting
9. Document the entire testing process and its results

Include Functional and Non-Functional Testing, Modeling and Simulations, challenges you have determined.

4.1 Unit Testing

– Discuss any hardware/software units being tested in isolation

4.2 Interface Testing

– Discuss how the composition of two or more units (interfaces) are to be tested. Enumerate all the relevant interfaces in your design.

4.3 Acceptance Testing

How will you demonstrate that the design requirements, both functional and non-functional are being met? How would you involve your client in the acceptance testing?

4.4 Results

– List and explain any and all results obtained so far during the testing phase

- Include failures and successes
- Explain what you learned and how you are planning to change the design iteratively as you progress with your project
- If you are including figures, please include captions and cite it in the text

5 Implementation

Describe any (preliminary) implementation plan for the next semester for your proposed design in 3.3.

6 Closing Material

6.1 Conclusion

Summarize the work you have done so far. Briefly reiterate your goals. Then, reiterate the best plan of action (or solution) to achieving your goals and indicate why this surpasses all other possible solutions tested.

6.2 References

List technical references and related work / market survey references. Do professional citation style (ex. IEEE).

6.3 Appendices

Any additional information that would be helpful to the evaluation of your design document.

If you have any large graphs, tables, or similar data that does not directly pertain to the problem but helps support it, include it here. This would also be a good area to include hardware/software manuals used. May include CAD files, circuit schematics, layout etc,. PCB testing issues etc., Software bugs etc.