A logo with a black background

Description automatically generated with medium confidence

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
| *Add date* | *Add date* | *Add date* |

|  |  |
| --- | --- |
| **Project:** | Park Effect |
| **Document type:** | Guide |
| **Document number:** | 23009-GU-0001 |
| **File name:** | 23009-GU-0001-Optimization Framework Guide |
| **Version / Status:** | 0.1 / Draft |
| **Last update:** | 14-8-2024 |
| **Maintainer:** | Chris Osse |
| **Authors:** | Chris Osse |
| **Reviewed by:** |  |
| **Signed by:** |  |

Guide

Optimization Framework

Confidentiality

THIS REPORT IS CONFIDENTIAL AND IT IS NOT ALLOWED TO SPREAD THIS REPORT TO ANY PERSON OR ORGANISATION OTHER THAN TOUCHWIND WITHOUT THE EXPLICIT AND WRITTEN PERMISSION OF A TOUCHWIND REPRESENTATIVE.

Contributing authors

|  |  |
| --- | --- |
| AUTHOR | PART(S) OF THIS DOCUMENT |
| **Chris Osse** | All chapters |

Document changes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| VERSION | STATUS | DESCRIPTION | DATE | Author(s) | REVIEWer(s) |
| 0.1 | Draft | Document creation | 14-08-2024 | Chris Osse |  |

Reference documents

|  |  |  |  |
| --- | --- | --- | --- |
| Ref Nr. | DOCUMENT NUMBER / AUTHOR | DOCUMENT TILTE | date |
| 1 | Chris Osse | Global Power Output Optimization of Tilted Wind Turbine Farms | 14-08-2024 |

# List of abbreviations

|  |  |
| --- | --- |
| Abbreviation | Written out |
| ABL | Atmospheric Boundary Layer |
| CCM | Cumulative Curl Misalignment (model) |
| FLORIS | FLOw Redirection and Induction in Steady State |
| LES | Large Eddy Simulation |
| SOWFA | Simulator fOr Wind Farm Applications) |

# List of symbols

|  |  |  |
| --- | --- | --- |
| Symbol | Unit | Description |
|  | *Type unit* | *Type description* |

Table of Contents

[Confidentiality 1](#_Toc172709869)

[Contributing authors 1](#_Toc172709870)

[Document changes 1](#_Toc172709871)

[Reference documents 1](#_Toc172709872)

[List of abbreviations 1](#_Toc172709873)

[List of symbols 1](#_Toc172709874)

[1 Chapter name 3](#_Toc172709875)

# Introduction

In this document, it is explained how to install and use the TouchWind Optimization Framework as established in the Graduation Project of Chris Osse [1]. All necessary code can be found by the following links:

* TouchWind Optimization Framework   
  <https://github.com/chrisosse/TouchWind_Optimization_Framework>   
  Includes the Framework, enabling to quickly add new low to mid fidelity models and compare them with reference data. Wind Farm cases can easily be created and modified, after which the Wind Farm can be optimized. Several visualization options are available.
* FLORIS with the Cumulative Curl Misalignment (CCM) model implementation   
  <https://github.com/chrisosse/floris_tilt>   
  Includes FLORIS v3.4.1 with the Cumulative Curl Misalignment (CCM) model implementation.
* All code used for the Graduation Project   
  <https://github.com/chrisosse/TouchWind-GPO-Optimization>   
  Includes a lot of code used for sensitivity studies, calibration and validation. It is unorganized, but if one needs some kind of visualization or an example of a piece of code used for before mentioned things, one can take a look in this repository.

For questions or remarks, one can contact Chris Osse via email and phone:

Email: [ossechris@gmail.com](mailto:ossechris@gmail.com)

Phone: +31 6 29 92 53 65

# Installation

In this chapter, it is explained how to install the framework and all other necessities. If one would like to make changes to the framework or other things, it is recommended to either fork the desired repositories

## TouchWind Optimization Framework

To use the framework, the GitHub repository has to be cloned locally. This is done by following the following steps:

1. Open a terminal, and navigate to the desired path to install the framework.
2. Enter the following command to clone the framework:   
   git clone <https://github.com/chrisosse/TouchWind_Optimization_Framework.git>
3. Create a new Python environment for the TouchWind Optimization Framework. It is highly recommended to use a Python virtual environment manager such as [conda](https://docs.conda.io/en/latest/miniconda.html) in order to maintain a clean and sandboxed environment ([www.anaconda.com/](https://www.anaconda.com/)), but any method can be used.
4. Enable the Python environment in the terminal.
5. Enter the following command to install all required Python packages:   
   pip install -r requirements.txt

## FLORIS CCM model

To be able to use the CCM model as implemented in FLORIS by Chris Osse, follow the following steps:

1. Open a terminal, and navigate to the desired path to install FLORIS.
2. Make sure the TouchWind Optimization Framework is enabled.
3. Enter the following command to clone FLORIS with the CCM model implementation:   
   git clone <https://github.com/chrisosse/floris_tilt.git>
4. Enter the following command to install the package in the Python environment:   
   pip install -e floris\_tilt

If one would like to make changed to FLORIS and enable these changes, the following steps must be followed:

1. Make the desired changes to FLORIS.
2. Open a terminal and enable the TouchWind Optimization Framework environment.
3. Go to the path where FLORIS with the CCM model implementation is installed.
4. Enter the following command to reinstall the package in the Python environment:   
   pip install -e floris\_tilt

## Extra code

If one would like to use all code used for the Graduation Project of Chris Osse as example for his or her own code, the following steps need to be followed:

1. Open a terminal, and navigate to the desired path to install the extra code.
2. Enter the following command to clone the extra code:   
   git clone <https://github.com/chrisosse/TouchWind-GPO-Optimization.git>

# How to use

The optimization framework includes one Python Jupiter Notebook file, in which several functions of the framework are used as an example. These functions can be used to setup ones own problems, cases and optimization requirements. In the notebook, it is explained in more detail how to use the functions; a quick overview of the options is given here:

* Add low to mid fidelity model   
  In the folder *\models*, a Python file named *TestModel.py* can be found. This file can be duplicated and altered to add a new model to the framework. The required functions with their input and output variables are already created, so only the functionality itself, which is dependent on the new model, has to be implemented.
* Add reference model   
  For reference data, also a new model can be added. For now, only LES is supported. When one wants to implement a new type of reference data, the *LES.py* file, found in *\models* can be duplicated and altered to allow new reference data types.
* Model choice   
  A choice for both the low to mid fidelity model, as well as the reference model, can easily be changed.
* Case manager   
  A case manager instance can be created, which can either contain cases as specified in an Excel file, as well as custom created cases. Each case consists of several variables, which are sorted in three categories:
  1. Conditions  
     Containing information about wind conditions such as wind speed, wind direction, turbulence and the Atmospheric Boundary Layer (ABL).
  2. Layout  
     Containing information about the Wind Turbine locations and size.
  3. Turbines  
     Containing information about individual Wind Turbine properties such as yaw, tilt and thrust coefficient.
* Run cases   
  To obtain both information about the individual Wind Turbine power outputs and about the resulting flow fields. These results can be plotted as well.
* Optimization  
  Optimize a case, where settings such as the parameters to optimize, their boundaries and the downwind values can be changed. Furthermore, optimization options such as stopping conditions can be altered as well. The results are saved in both an Excel file and can optionally be saved in a text file, which can easily be used to copy resulting values into simulations such as LES (SOWFA).
* Optimization analysis   
  Analysis of the optimization procedure and the results, where the resulting power over wind direction can be visualized, next to the amount of iterations and model runs per wind direction.
* Reference data   
  Data of reference models such as LES can be obtained. This includes data about the individual Wind Turbine power outputs, as well as flow field data. This can also be visualized.