Announcing FLORIS Version 3.2 and FLASC Version 1.1

The National Renewable Energy Laboratory (NREL) has just launched a new version of two wind energy tools. The updates include version 3.2 of FLOw Redirection Induction in Steady State (FLORIS) and version 1.1 of FLORIS-based Analysis for SCADA (FLASC). The updates increase the tools' capabilities and user experience.

The following details these updates and explains how to upgrade the tools on your computer. You are also invited to engage with the community on GitHub to support continued improvements to these tools.

FLORIS Version 3.2

FLORIS Version 3.2 (v3.2) is now available. This update includes a number of important corrections and improvements to FLORIS version 3.0.

Updates to FLORIS in v3.2 include:

Updates to FLORIS in v3.2 include:
- Refactoring layout optimization routines to improve modularity and performance across different objectives and optimization methods
- Adding turbine weighting to power and AEP calculations to improve user flexibility and computational performance
- Expanding functionality of the `UncertaintyInterface` to mirror the `FlorisInterface`
- Adding wind rose and power rose tools from previous versions of FLORIS
- Enabling time series calculations in a vectorized form
- Support for modeling rotors with a single grid-point for increased computational speed
- Adding methods to extract turbulence intensities and wind speeds at turbines
- Adding reference air density parameters to turbine performance data
- Fixing bugs and correcting capabilities to:
  - Enable heterogenous maps to be copied; this is important for the Serial-Refine optimization method
  - Raise an error if analysis routines are used prior to running the wake calculation (i.e. `calculate_wake`)
  - Enforce a maximum number of points on a turbine: 9 points in a 3x3 grid
  - Plot turbine positions on horizontal slices with `visualization.plot_turbines_with_fi`
  - Improve API for turbine layout definitions in `FlorisInterface.reinitialize.`

FLASC Version 1.1

The open-source [FLASC](#) companion repository implements the methods used in NREL's research for field data postprocessing, model analysis, and FLORIS-to-SCADA data comparison. It is now in Version 1.1 with the introduction of git-based code development practices matched to those used by FLORIS. As with FLORIS we welcome submissions of bug reports and feature requests to the [issues](#) page, as well as questions posted to [discussions](#).

How To Upgrade FLORIS

If you have not yet upgraded from FLORIS v2 to v3, see the instructions in the installation documentation.

If you have FLORIS installed with PyPI or Conda, you can upgrade to the latest version using your package manager, by copying and pasting the corresponding line of the following code into a terminal:

```
<code>
conda update floris
pip install floris--upgrade
</code>
```

Alternatively, if you've installed FLORIS by cloning the repository, then you can upgrade by pulling the latest release. Copy and paste the following code into a terminal:

```
<code>
cd floris/
git fetch --all
git stash  # if you've made uncommitted changes to the
source code
git pull origin main  # "origin" should be replaced by the
appropriate remote
git stash pop  # restores any uncommitted changes stash
above

# Install locally
pip install -e .
</code>
```

How To Upgrade FLASC

FLASC is not yet available in Conda or PyPI, so the local upgrade method is the only currently available option.

To upgrade locally, copy and paste the following code:

```
<code>
cd flasc/
git fetch --all
git stash  # if you've made uncommitted changes to the
source code
git pull origin main  # "origin" should be replaced by
the appropriate remote
git stash pop  # restores any uncommitted changes
stash above

# Install locally
pip install -e .
</code>
```

How To Engage With the FLORIS and FLASC Community

FLORIS leverages the following features of GitHub to coordinate support and development efforts:

- Discussions. Collaborate to develop ideas for new use cases, features, and software designs and get support for usage questions
    - Ideas is a great spot to develop the details for a feature request
    - Q&A is the place for user support
    - Show off things you're doing with FLORIS in Show and Tell
- Issues. Report potential bugs and well-developed feature requests
- Projects. Include current and future work on a timeline and assign a person to "own" it.

Guidelines for interacting with the repository with git and GitHub are available in the developer documentation.

If you have additional input, no longer wish to receive updates on FLORIS, or just discovered FLORIS and would like to start receiving updates, please send an email to Paul.Fleming@nrel.gov.