Delft University of Technology
Faculty of Electrical Engineering, Mathematics and Computer Science
Department of Electrical and Sustainable Energy
Group of DC Systems, Energy Conversion and Storage

User Manual

csvreader.py

Author:
Dr. Joel Alpízar-Castillo

Version 1.0
31 July, 2025

Contents

Chapter 1

Introduction

This document provides a description of the functions used to simulate the residential multi-carrier energy systems, together with a low-voltage distribution networks in the dissertation [1]. Each chapter provides a description of each of the functions used, including a brief mathematical explanation and details of the arguments. In addition, example files are also provided.

Note that all the functions and example files were tested in Sypder 5.5.1, using Python 3.12.7, and Windows 10.

The FLEXINet project was carried out with a Top Sector Energy subsidy from the Ministry of Economic Affairs and Climate, carried out by the Netherlands Enterprise Agency (RVO). The specific subsidy for this project concerns the MOOI subsidy round 2020.

Chapter 2

csvreader

This library was not created for the FLEXINet project, but as part of the MSc thesis [2]. Nevertheless, as it is constantly used, its functions are also detailed. The library itself is stored in the repository in [3].

For this library to be used, the following module must be installed:

- itertools
- pandas
- numpy

2.1 read_data Class

The `read_data` class provides a lightweight interface for reading and processing CSV files without relying on full-featured data frames. It allows conversion of CSV data into lists of columns, rows, matrices, or arrays.

Constructor

```
read_data(csv='', delim=';', address='CSVs\\')
```

Arguments:

- **csv [str]:** Name of the CSV file, including extension (e.g., `"data.csv"`).
- **delim [str, optional]:** Delimiter used in the CSV file (default is `';'`).
- **address [str, optional]:** Path to the folder containing the CSV file (default is `'CSVs\'`).

Attributes:

- **df:** The raw data frame read from the CSV.
- **nrows:** Number of rows in the data.
- **ncols:** Number of columns in the data.

Methods

data2cols()

Converts the data frame into a list of columns. Each column is stored as a list in `self.col`.

data2rows()

Converts the data frame into a list of rows. Each row is stored as a list in `self.row`.

data2matrix()

Converts the data frame into a NumPy matrix and stores it in `self.mat`.

data2array()

Converts the data frame into a NumPy array and stores it in `self.ar`.

Bibliography

[1] J. Alpízar-Castillo, "Residential multi-carrier energy storage systems as potential flexibility providers in low-voltage networks: A new player has joined the game," Ph.D. dissertation, Delft University of Technology, 2025. [Online]. Available: https://repository.tudelft.nl/record/uuid:76075049-7656-4c49-b206-e742a65b6062.

[2] J. Alpízar-Castillo, "Model predictive control implementation for the ocean grazer wave energy converter with a port-hamiltonian model," M.S. thesis, University of Groningen, Costa Rica Institute of Technology, 2018. [Online]. Available: https://repositoriotec.tec.ac.cr/handle/2238/10439.

[3] J. Alpízar-Castillo, *CSV Python Tools*, https://github.com/jjac13/CSV_Python_Tools, 2022.