demandlib Documentation

Release beta

Uwe Krien, oemof developing group

Jan 27, 2021

Contents

Contents:

CHAPTER 1

Getting started

The **demandlib** is part of the oemof group but works as a standalone application.

Table of contents

1.1 Introduction

With the demandlib you can create power and heat profiles for various sectors by scaling them to your desired demand. Additionally you can specify a year so that holidays are considered respectively.

1.2 Current Release

Install the demandlib using pypi:

```
pip install demandlib
```

You can also install the beta version with the most actual changes:

```
pip install git+https://github.com/oemof/demandlib
```

1.3 Developing Version

1.3.1 Documentation

Read the latest documentation at Readthedocs.org

http://demandlib.readthedocs.org

1.3.2 Installation

Clone the demandlib from github.

```
git clone git@github.com:oemof/demandlib.git
```

If the project is cloned you can install it using pip with the -e flag.

```
pip install -e <path/to/the/demandlib/root/dir>
```

1.3.3 Developing the demandlib

As the demandlib is part of the oemof developer group we use the same developer rules:

http://oemof.readthedocs.io/en/stable/developing_oemof.html

If you have push rights clone this repository to your local system.

```
git clone git@github.com:oemof/demandlib.git
```

If you want to contribute, fork the project at github, clone your personal fork to your system and send a pull request.

1.4 Example

We provide two examples on how to use the demandlib. One for the heat sector, executable by calling `demandlib_heat_example`, and one showing how to create electricity demand time series, executable by calling `demandlib_power_example`. Both examples are callable from anywhere in the command-line.

CHAPTER 2

What's New

These are new features and improvements of note in each release

CHAPTER 3

Model Description

3.1 BDEW

Using the demandlib you can create heat and electrical profiles by scaling the BDEW profiles to your desired annual demand. The BDEW profiles are the standard load profiles from BDEW.

3.1.1 Heat Profiles

Heat profiles are created according to the approach described in the BDEW guideline.

The method was originally established in this PhD Thesis at TU Munich.

The approach for generating heat demand profiles is described in section 4.1 (Synthetic load profile approach).

$$Q_{day}(\theta) = KW \cdot h(\theta) \cdot F \cdot SF$$

KW: Kundenwert (customer value). Daily consumption of customer at $\approx 8°C$, depending on SLP type and Temperature timeseries.
h: h-Wert (h-value) , depending on SLP type and daily mean temperature.
F: Wochentagsfaktor (week day factor), depending on SLP type and day of the week.
T: Daily mean temperature 2 meters above the ground (simple mean or "geometric series", which means a weighted sum over the previous days).
SF: Stundenfaktor (hour factor)

The geometric series approach is meant to account for thermal inertia.

$$\theta = \frac{T_t + 0.5 \cdot T_{t-1} + 0.25 \cdot T_{t-2} + 0.125 \cdot T_{t-3}}{1 + 0.5 + 0.25 + 0.125}$$

Depending on the profile type, different coefficients A, B, C, D for the sigmoid function are used.

$$h(\theta) = \frac{A}{1 + (\frac{B}{\theta - \theta_0})^C} + D$$

$$\theta_0 = 40°C$$

Types of houses:

EFH: Einfamilienhaus (single family house)
MFH: Mehrfamilienhaus (multi family house)
GMK: Metall und Kfz (metal and automotive)
GHA: Einzel- und Großhandel (retail and wholesale)
GKO: Gebietskörperschaften, Kreditinstitute und Versicherungen (Local authorities, credit institutions and insurance companies)
GBD: sonstige betriebliche Dienstleistung (other operational services)
GGA: Gaststätten (restaurants)
GBH: Beherbergung (accommodation)
GWA: Wäschereien, chemische Reinigungen (laundries, dry cleaning)
GGB: Gartenbau (horticulture)
GBA: Backstube (bakery)
GPD: Papier und Druck (paper and printing)
GMF: haushaltsähnliche Gewerbebetriebe (household-like business enterprises)
GHD: Summenlastprofil Gewerbe/Handel/Dienstleistungen (Total load profile Business/Commerce/Services)

3.1.2 Electrical Profiles

The electrical profiles are the standard load profiles from BDEW. You can choose from H0, G0...G6 and L0...L3 (?) by defining an annual demand for any of these options.

3.2 Further Profiles

We implemented further profiles (one until now) to represent further demand sectors which are not covered by the BDEW load profiles.

3.2.1 Industrial Electrical Profile

The industrial electrical profile uses a step function.

CHAPTER 4

demandlib package

4.1 Submodules

4.2 demandlib.bdew module

Load profiles

SPDX-FileCopyrightText: Patrik Schönfeldt

SPDX-License-Identifier: MIT

4.3 demandlib.particular_profiles module

Implementation of the bdew standard load profiles for electric power.

class demandlib.particular_profiles.**IndustrialLoadProfile**(*dt_index*, *holidays=None*)

Bases: object

Generate an industrial heat or electric load profile.

simple_profile(*annual_demand*, ***kwargs*)

Create industrial load profile

Parameters **annual_demand** (*float*) – Total demand.

Other Parameters

- **am** (*datetime.time*) – beginning of workday
- **pm** (*datetime.time*) – end of workday
- **week** (*list*) – list of weekdays
- **weekend** (*list*) – list of weekend days

- **profile_factors** (*dictionary*) – dictionary with scaling factors for night and day of weekdays and weekend days

4.4 demandlib.tools module

Tools needed by the main classes

demandlib.tools.**add_weekdays2df**(*time_df*, *holidays=None*, *holiday_is_sunday=False*)

Giving back a DataFrame containing weekdays and optionally holidays for the given year.

Parameters

- **time_df** (*pandas DataFrame*) – DataFrame to which the weekdays should be added
- **Parameters** (*Optional*) –
- **----------** –
- **holidays** (*array with information for every hour of the year, if holiday or*) – not (0: holiday, 1: no holiday)
- **holiday_is_sunday** (*boolean*) – If set to True, all holidays (0) will be set to sundays (7).

Returns **pandas.DataFrame**

Return type DataFrame with weekdays

Notes

Using Pandas > 0.16

4.5 Module contents

CHAPTER 5

Indices and tables

- genindex
- modindex
- search

Python Module Index

d

Index