emerge-res-integration-tool

The emerge-res-integration-tool project is a wrapper implementing advanced power grid calculation features for the needs of EMERGE project, based on PowSyBl-Metrix tool.

License

Authors and acknowledgment

Developed within the EMERGE project by Artelys. Authors: Léo Rossignol, Alexis Godefroy

Description

The emerge-res-integration-tool implements the EMERGE methodology for integrating and optimising Battery Energy Storage Systems (BESS) within electrical networks. It relies on a coupled optimization process involving iterations between **PowSyBl-Metrix** and **Knitro** to optimize the dispatch (charging / discharging plan) of a given battery (or set of batteries) over any timeseries with a hourly timestep. Batteries are modelled as controllable assets in PowSyBl-Metrix, their setpoints (dispatch decisions) being fully handled by Knitro optimizer. The objective function consists of minimizing total system costs, including adequacy and redispatching costs computed by PowSyBl-Metrix, while respecting network and battery operational constraints (ratings, min and max charging / discharging powers, State of Charge).

Integration within EMERGE project

Designed to receive battery placement as input (substation ID), the tool supports iterative interactions with the external Optimal Storage Placement Tool (developed by CIRCE within the EMERGE project) to jointly optimize battery placement and operational usage.

Installation

Requirements

- Python >= 3.12
- The metrix-launcher package embedding PowSyBl-Metrix tool
- A valid license of Knitro solver
- Python dependencies listed in `requirements.txt`

It is recommended to use a dedicated virtual environment.

Usage

Inputs

The tool is executed by providing: - An input folder containing the network model (IIDM format), timeseries (csv file containing generation and load setpoints on the network, hourly timestep), configuration and mapping files (groovy files needed by PowSyBl-Metrix); - A list of nodes (IDs of substations) where batteries should be placed; - Optional sizing and cost parameters for batteries.

Process

The main optimisation workflow runs an iterative loop combining PowSyBl Metrix OPF calculations and Knitro optimisation until convergence or a maximum number of iterations is reached. Results are exported as CSV files, including battery dispatch profiles, battery state of charge, system costs (adequacy and redispatching), cost savings and avoided curtailment.

Outputs

The tool generates, among others: - Optimised battery power and state-of-charge profiles; - Adequacy and redispatching costs; - Cumulated system cost savings and curtailment avoided.

Quick Start

This section provides a quick start guide for users who wish to run the emerge-res-integration-tool. Users are expected to be comfortable with basic Python usage, including running scripts, managing environments, and interpreting logs. Finally, all input data (network, timeseries, configuration, and mapping files) must be properly prepared and consistent, as described in the previous sections of this documentation.

Before executing EMERGE, users must ensure that their working environment is properly configured. It is strongly recommended to use a dedicated Python >=3.12 virtual environment (such as venv or conda) in order to manage dependencies cleanly and avoid conflicts with other projects.

All required dependencies must be installed beforehand, typically via the requirements.txt file. In addition, if the optimization relies on external solvers (such as Knitro), these must be correctly installed, configured, and licensed.

A Python script quickstart.py is provided for quick start with the tool.

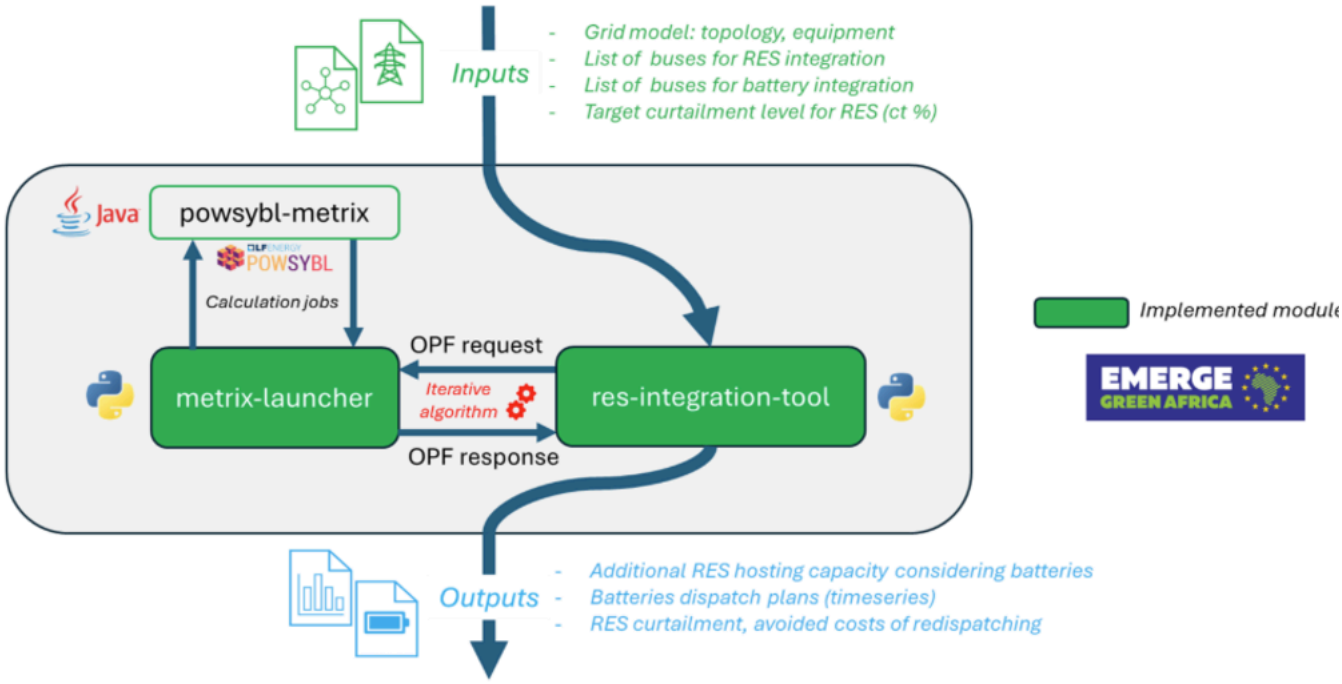
At the beginning of the process, a baseline battery profile is used as an initial input. The workflow then follows an iterative loop in which this profile is successively refined. At each iteration, the current battery profile is sent to the Optimal Storage Placement tool, which returns a list of optimal locations (buses) for battery placement. A battery is then placed on the best location, as input for the next optimization, which will compute an updated battery dispatch plan (charging / discharging operation profile). This updated profile is reused in the next iteration, creating a feedback loop between OSP tool (CIRCE) and emerge-res-integration-tool (Artelys).

The process continues until either: * The battery location is validated by OSP tool, or * A maximum number of iterations (10) is reached. Throughout the process, key results are stored at each iteration, including: *The battery profiles used as inputs ;* The list of substations recommended by OSP tool.

At the end of the execution, all results are exported into a single CSV file, providing a complete track of the iterative process for analysis and validation purposes. The output file of optimization process is also relevant to understand the operation and benefits of the tool.

Visuals

The workflow of this calculation tool is described below.



EMERGE workflow