Workshop

PyPSA: an open-source python environment for state-of-the-art energy system modelling

Dr. Iegor Riepin
Bobby Xiong
(Technische Universität Berlin)

4 TSOs & TU Berlin ENSYS

12 May 2025

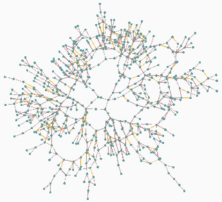
People: https://pypsa.org/
Docs: github.com/pypsa/pypsa-eur
Slide deck credits: https://fneum.org/

What is PyPSA?

Our research focus:

- Cost-effective pathways to reduce greenhouse gas emissions
- Evaluation of grid expansion, hydrogen strategies, carbon management strategies
- Co-optimisation of generation, storage, conversion and transmission infrastructure
- Algorithms to improve the tractability of models
- All open source and open data

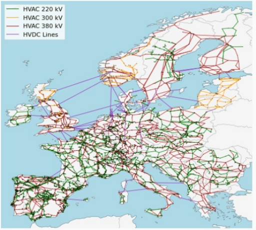
PyPSA



A python software toolbox for simulating and optimising modern power systems.

Documentation »

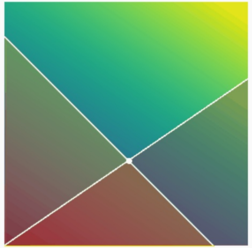
PyPSA-Eur



A Sector-Coupled Open Optimisation Model of the European Energy System

Documentation »

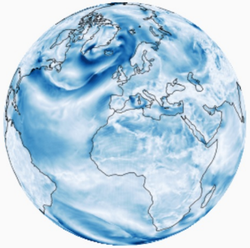
Linopy



Linear optimization interface for N-D labeled variables.

Documentation »

Atlite



A Lightweight Python Package for Calculating Renewable Power Potentials and Time Series

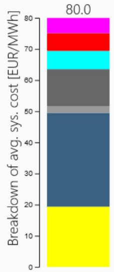
Documentation »

Powerplantmatching

A toolset for cleaning, standardizing and combining multiple power plant databases.

Documentation »

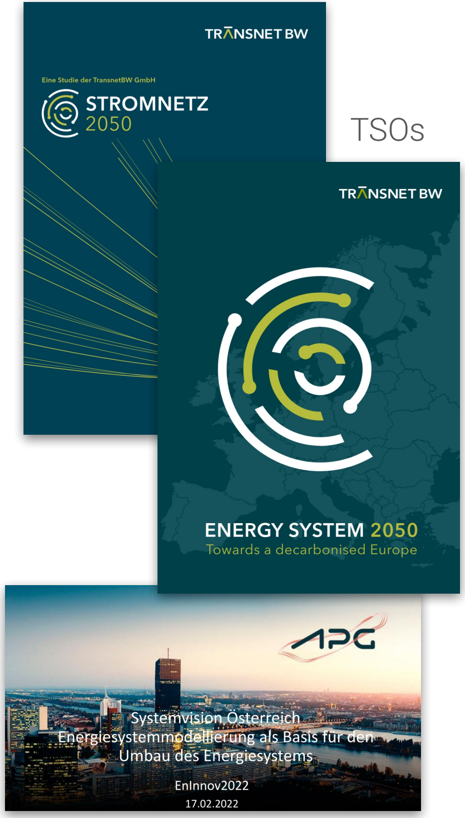
Model Energy



An online toolkit for calculating renewable electricity supplies.

Application examples

TSOs



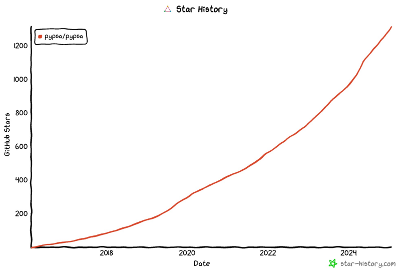
NGOs and international organisations

Regulators

International

https://pypsa.readthedocs.io/en/latest/references/users.html

PyPSA:
Python for Power System Analysis



Capabilities

Capacity expansion (linear)
- single-horizon
- multi-horizon

Market modelling (linear)
- Linear optimal power flow
- Security-constrained LOPF
- Unit commitment
- Dispatch & redispatch

Non-linear power flow
- Newton-Raphson

With components for

- Electricity transmission networks and pipelines.
- Generators with unit commitment constraints
- Variable generation with time series (e.g. wind and solar)
- Storage with efficiency losses and inflow/spillage for hydro
- Conversion between energy carriers (PtX, CHP, BEV, DAC)

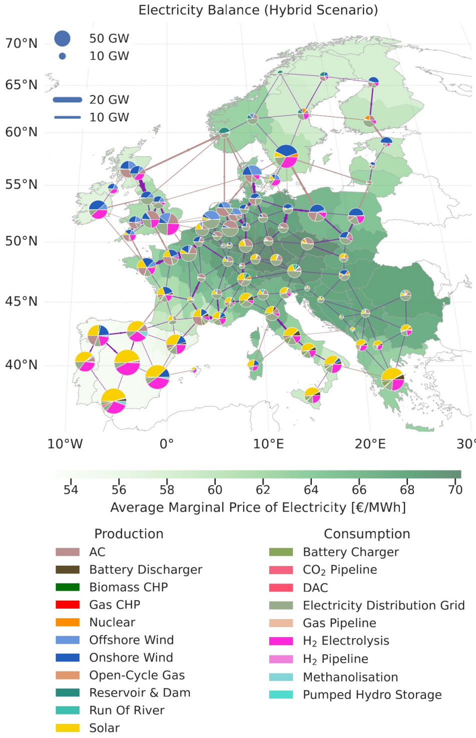
Backend

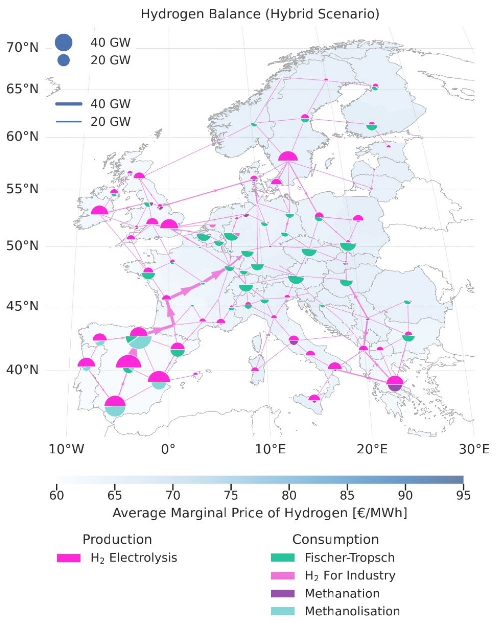
- all data stored in pandas
- framework built for performance with large networks and time series
- interfaces to major solvers (Gurobi, CPLEX, HiGHS, Xpress), with linopy (by PyPSA devs)
- highly customisable, but no GUI
- Suitable for greenfield, brownfield & pathway studies

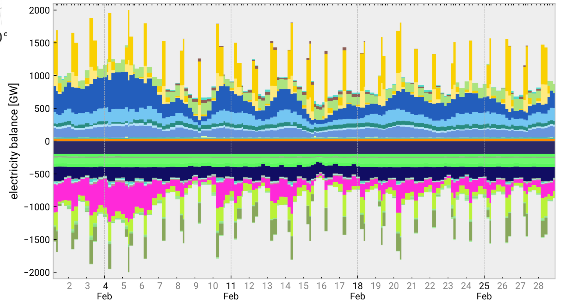
PyPSA-Eur: A sector-coupled open model of the European energy system

Automated workflow to build energy system model of Europe from raw open data with high spatial and temporal resolution:

1. OSM transmission lines (>220 kV) + TYNDP
2. a database of existing power plants,
3. time series for electricity demand,
4. time series for wind/solar availability, and
5. geographic wind/solar potentials
6. cost and efficiency assumptions
7. methods for model simplification
8. more for sector-coupled networks like pipelines, LNG terminals, electric vehicles, industry locations, … *(later)*







Energy infrastructure planning in PyPSA as an optimisation problem

Find the long-term cost-optimal energy system, including investments and short-term costs:

$$\text{Min}\left[\begin{array}{c}\textbf{Yearly}\\ \textbf{system costs}\end{array}\right] = \text{Min}\left[\sum_n \left(\begin{array}{c}\textbf{Annualised}\\ \textbf{capital costs}\end{array}\right) + \sum_{n,t}\left(\begin{array}{c}\textbf{Marginal}\\ \textbf{costs}\end{array}\right)\right]$$

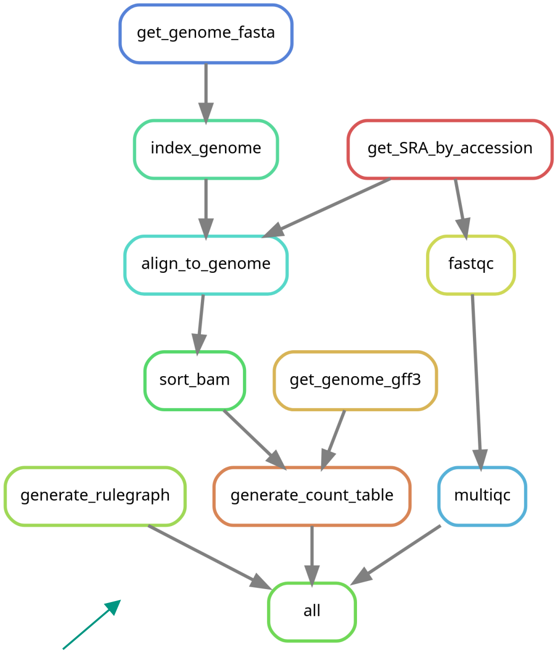
subject to

- meeting energy demand at each node n (e.g. region) and time t (e.g. hour of year) ⟸ data
- transmission constraints between nodes and linearised power flow ⟸ data
- wind, solar, hydro (variable renewables) availability time series $\forall\, n, t$ ⟸ data
- installed capacity $\leq$ geographical potentials for renewables ⟸ data
- fulfilling CO_2 emission reduction targets
- Flexibility from gas turbines, battery/hydrogen storage, HVDC links — More on that later!

Challenges with data-driven modelling

Create a full pipeline of data processing from raw data to results.

- Many different data sources
- Many data sources need cleaning and processing
- Many intermediate scripts and datasets
- Data and software dependencies need to be managed
- Data and code change over time
- Want to be able to reproduce results
- Want to run many different scenarios



Requires a scalable workflow management tool! → snakemake

Originally comes from bioinformatics field.

Miniature example of snakemake

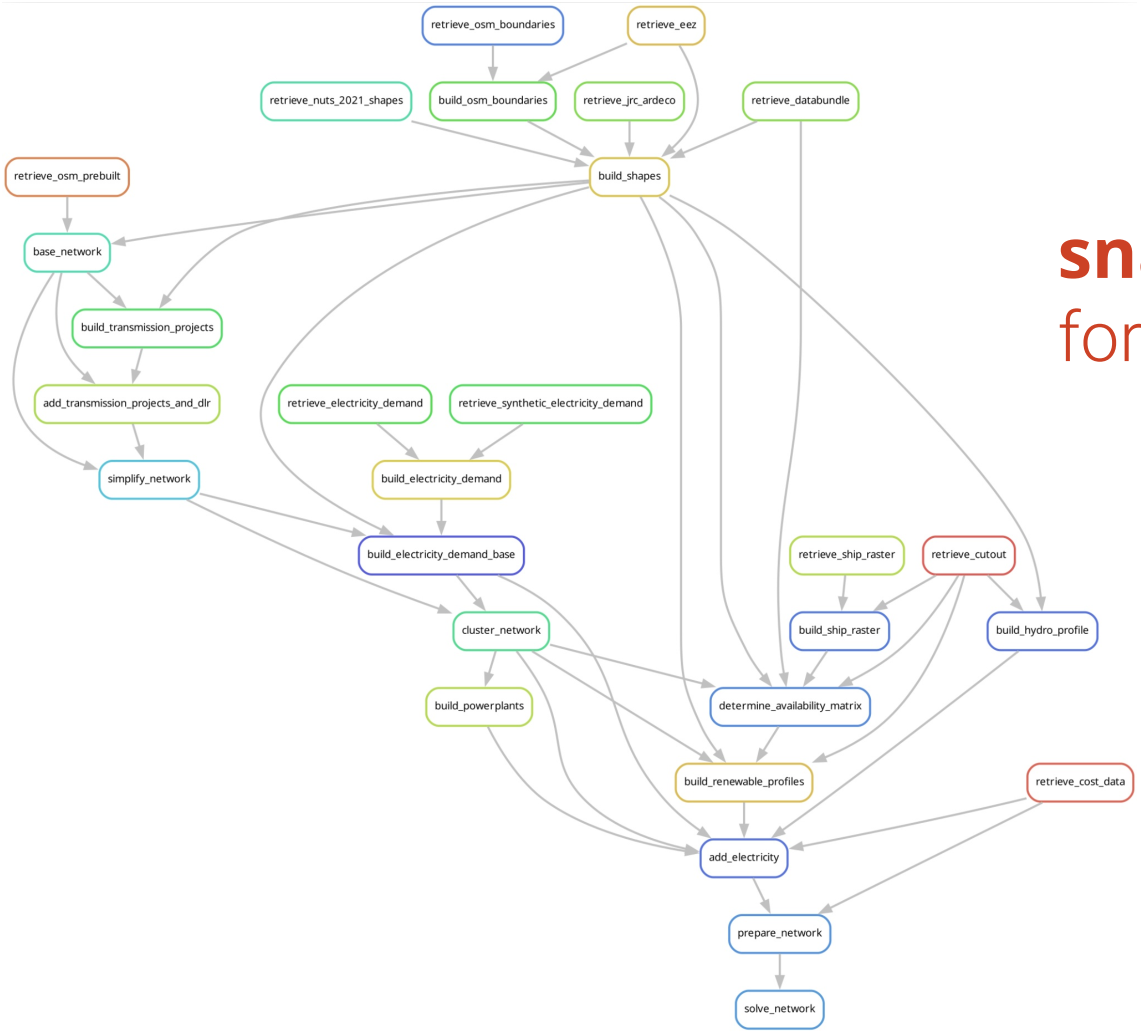
Snakefile

```
rule mytask:
    input:
        "data/{sample}.txt"
    output:
        "result/{sample}.txt"
    script:
        "scripts/mytask.py"
```

```
rule myplot:
    input:
        "result/{sample}.txt"
    output:
        "figures/{sample}.pdf"
    script:
        "scripts/myplot.py"
```

command: `$ snakemake figures/myfigure.pdf`

Example adapted from Johannes Köster,
https://slides.com/johanneskoester/snakemake-short#/



snakemake workflow
for the electricity sector

Simplified workflow structure

Automated **downloads**

Process the data

Build the electricity system model

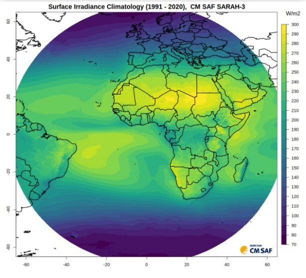
Extend the model by other sectors (later)

Solve, summarize & visualize the results

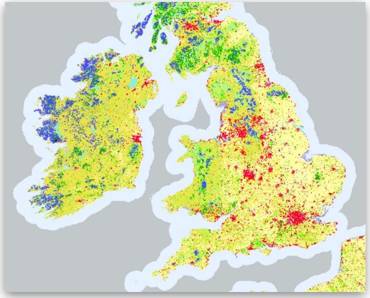
First, raw data is automatically downloaded.

WDPA

SARAH-3

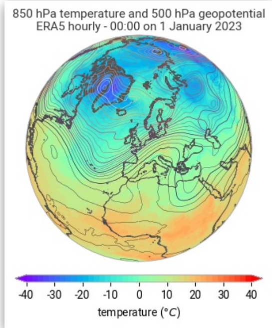


CORINE

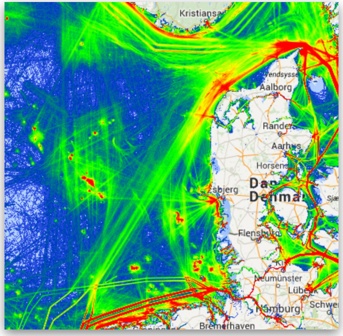


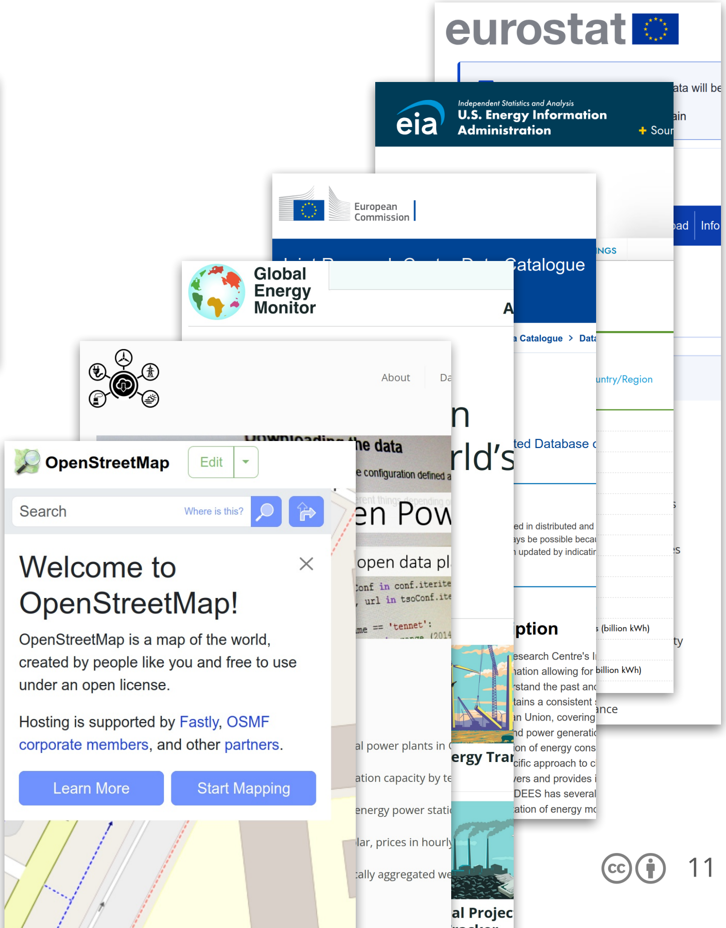
GEBCO

ERA5



World Bank





https://pypsa-eur.readthedocs.io/en/latest/data_sources.html

Simplified workflow structure

Automated **downloads**

Process the data

Build the electricity system model

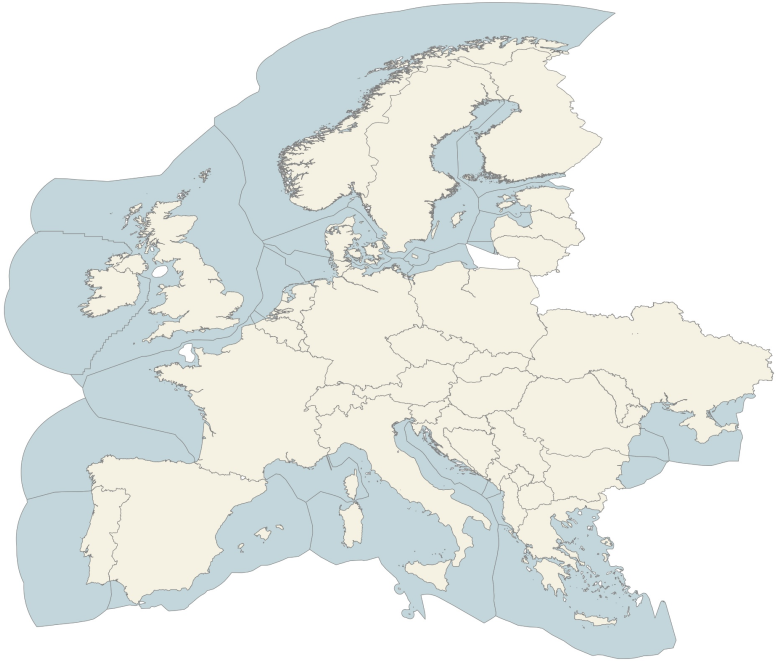
Extend the model by other sectors (later)

Solve, summarize & visualize the results

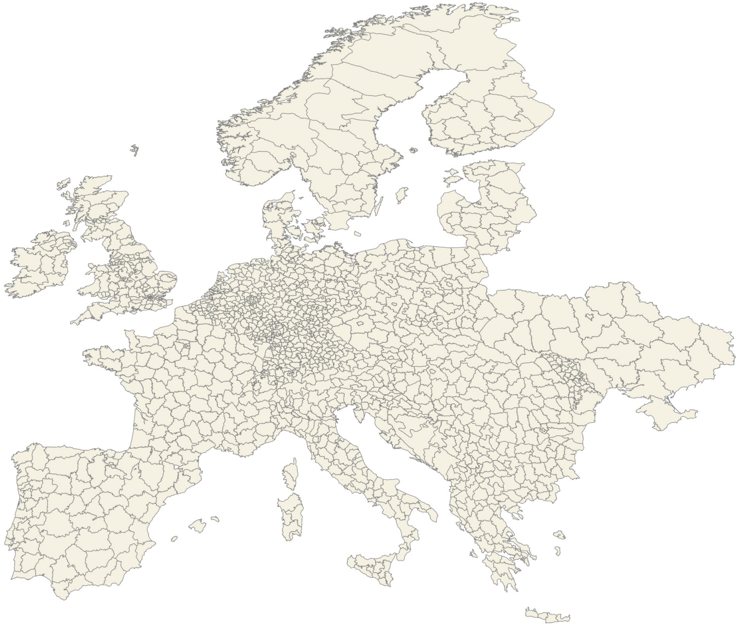
Steps to building PyPSA-Eur electricity system

Retrieve onshore & offshore polygons for each country	`build_shapes`

Country shapes & exclusive economic zones (EEZ)



NUTS administrative regions (NUTS3)

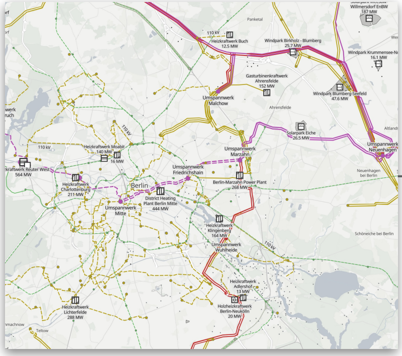


Steps to building PyPSA-Eur electricity system

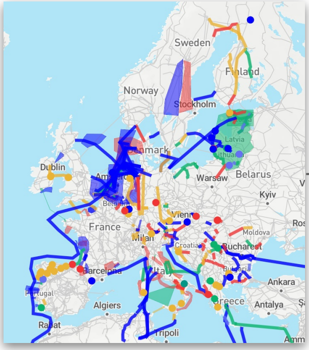
Retrieve onshore & offshore polygons for each country	`build_shapes`
Construct a base high-voltage network with buses, transformers, AC & DC lines with DLR & TYNDP	`base_network`, `build_transmission_projects`

Power grid topology

OpenStreetMap data



TYNDP projects



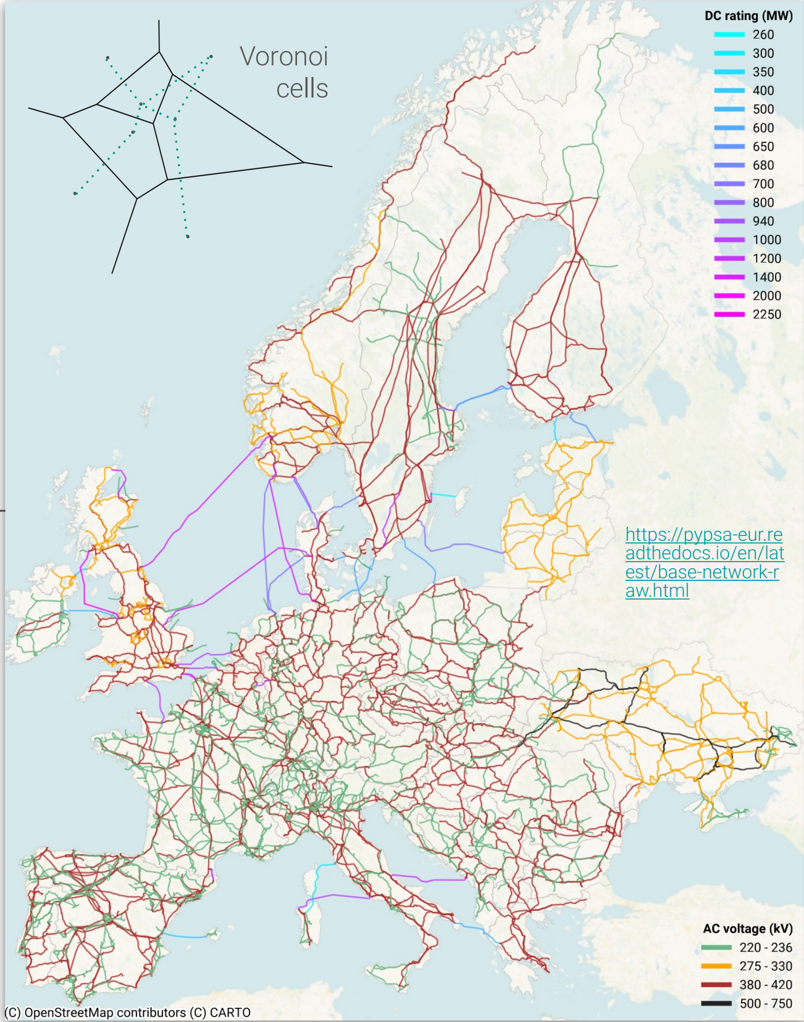
Apply **standard line types** for capacity and parameters.

Calculate **dynamic line rating** potential from weather data.

European network with

- ~5,800 buses
- ~7,300 AC lines (>220 kV)
- 36 HVDC links (+TYNDP)

https://www.nature.com/articles/s41597-025-04550-7



https://pypsa-eur.readthedocs.io/en/latest/base-network-raw.html

Steps to building PyPSA-Eur electricity system

Retrieve onshore & offshore polygons for each country	`build_shapes`
Construct a base high-voltage network with buses, transformers, AC & DC lines with DLR & TYNDP	`base_network`, `build_transmission_projects`
Transform all transmission lines to 380kV, remove dead ends & cluster with k-means or hierarchical clustering	`simplify_network`, `cluster_network`

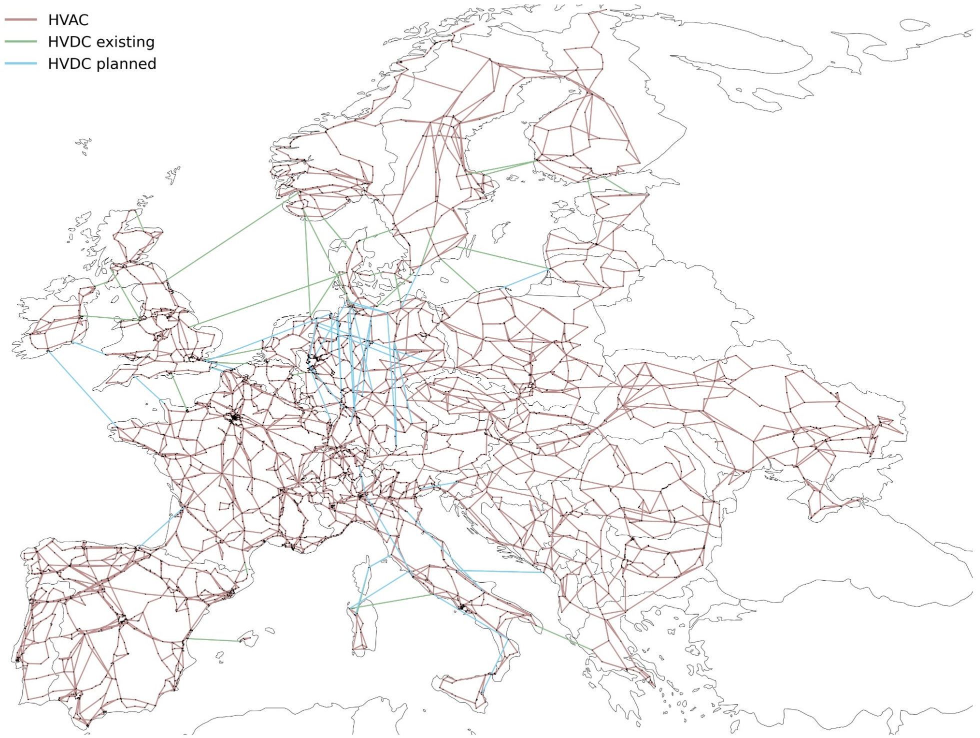
Clustering the electricity network: `simplify_network`

Need to make the optimization problem less **computationally challenging**…

…if we want to **co-optimize** generation, storage, PtX conversion and transmission infrastructure:

1. Lift all lines to common voltage level of 380 kV.

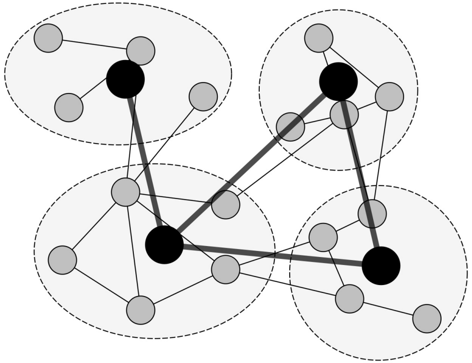
2. Remove dead ends.

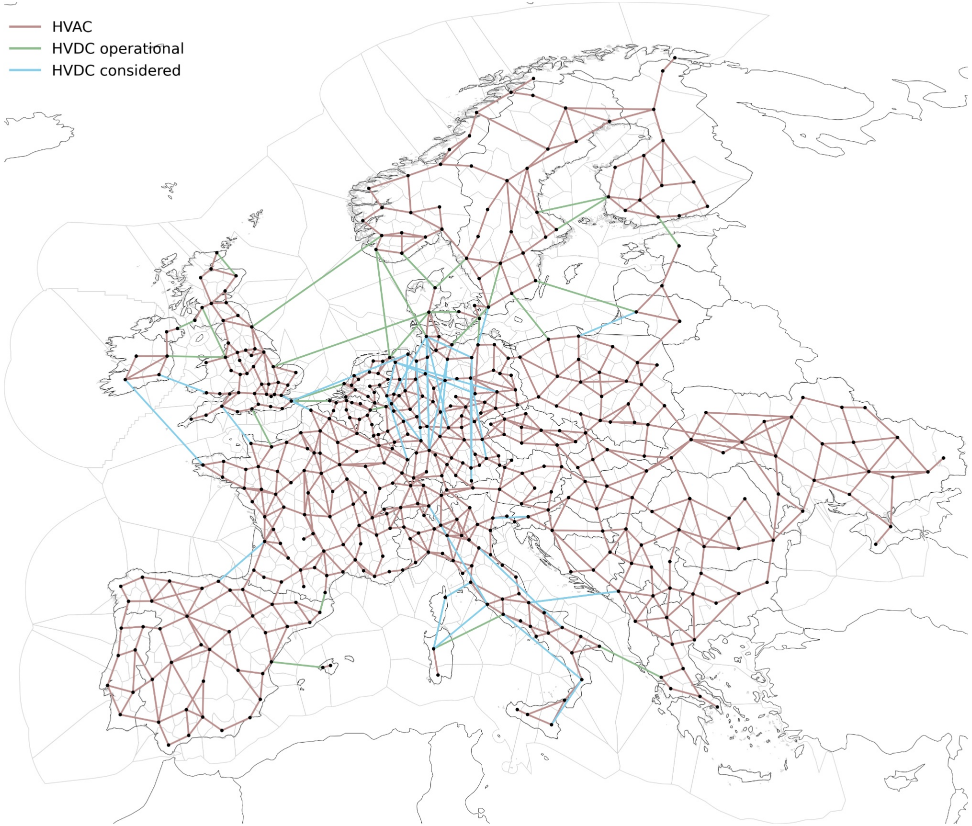


Clustering the electricity network: `cluster_network`

Transformed
to 380 kV

Clustered to
512 regions

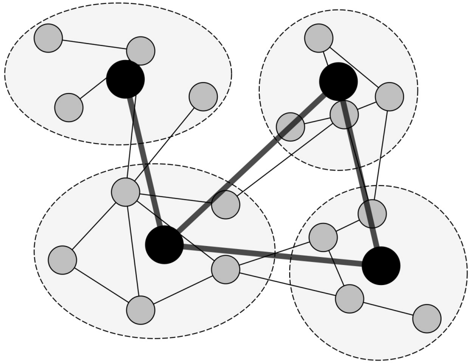


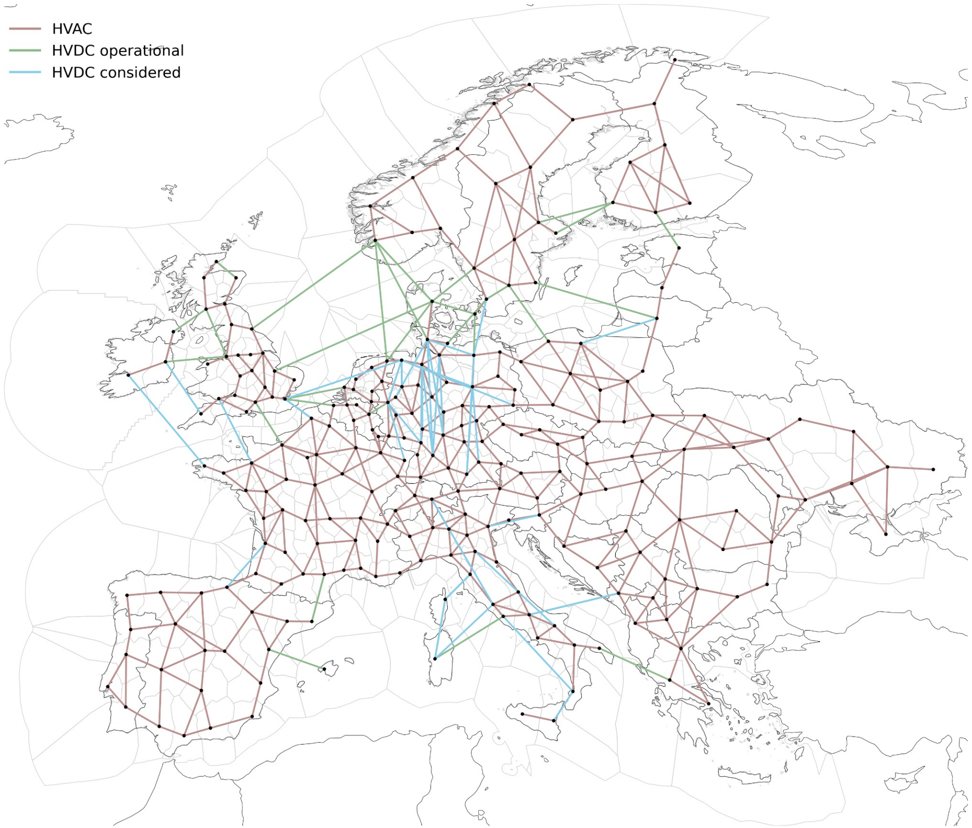


Clustering the electricity network: `cluster_network`

Transformed
to 380 kV

Clustered to
256 regions

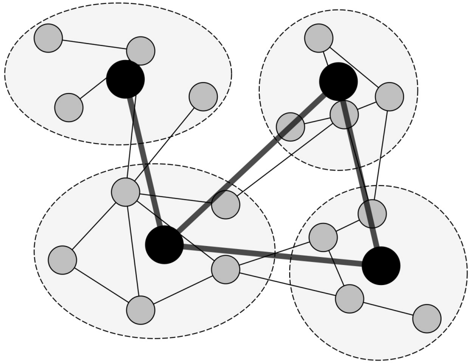




Clustering the electricity network: `cluster_network`

Transformed **to 380 kV**

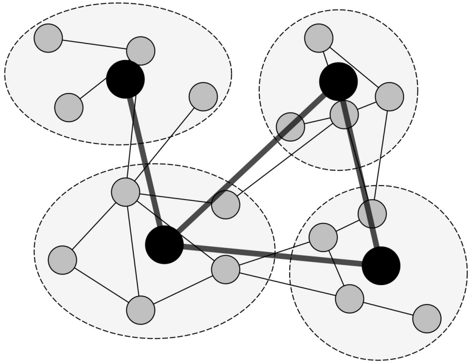
Clustered to **128 regions**



Clustering the electricity network: `cluster_network`

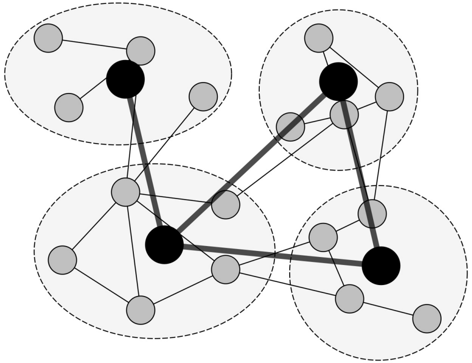
Transformed
to 380 kV

Clustered to
64 regions



Clustering the electricity network: `cluster_network`

Transformed **to 380 kV**	Clustered to **41 regions**



Steps to building PyPSA-Eur electricity system

Retrieve onshore & offshore polygons for each country	`build_shapes`
Construct a base high-voltage network with buses, transformers, AC & DC lines with DLR & TYNDP	`base_network`, `build_transmission_projects`
Transform all transmission lines to 380kV, remove dead ends & cluster with k-means or hierarchical clustering	`simplify_network`, `cluster_network`
Determine eligible areas for utility-scale PV & onshore/offshore wind park development	`determine_availability_matrix`
Build renewable capacity factor profiles for each clustered region based on land availability	`build_renewable_profiles`, `build_hydro_profile`

atlite: Convert weather data to energy systems data

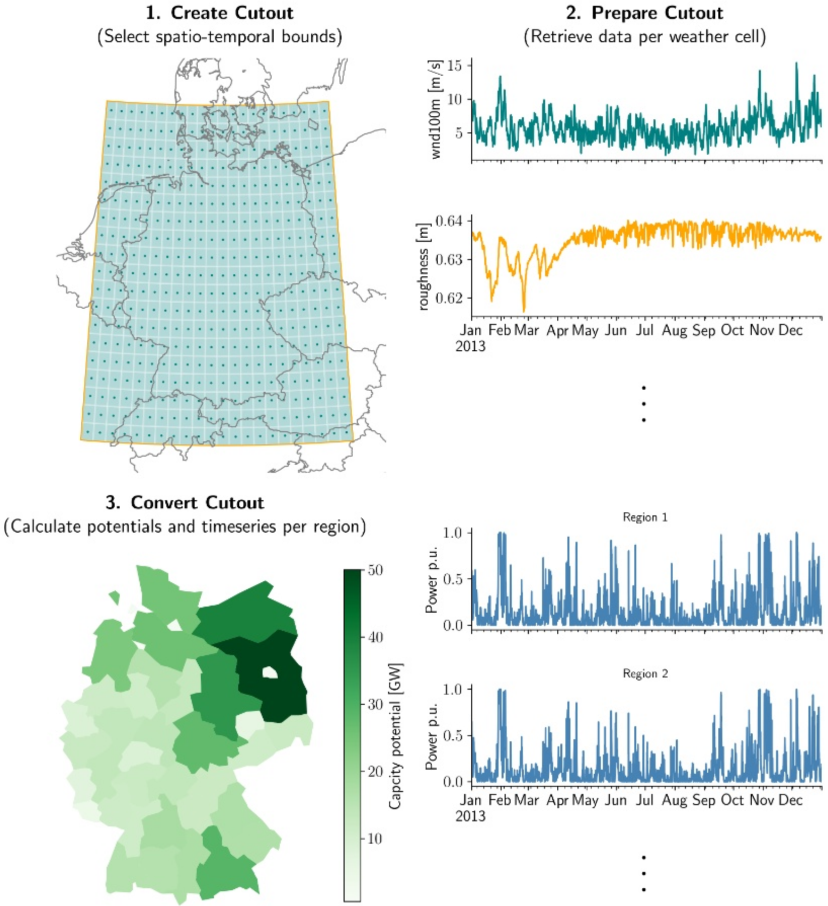
pypi v0.3.0 | conda-forge v0.3.0 | Tests passing | codecov 72% | docs passing | license MIT
REUSE compliant | JOSS 10.21105/joss.03294 | chat 52 online | stackoverflow pypsa questions 44

Python library for converting **weather data** (e.g. wind, solar radiation, temperature, precipitation) into **energy systems data**:

- solar photovoltaics
- solar thermal collectors
- wind turbines
- hydro run-off, reservoir, dams
- heat pump COPs
- dynamic line rating (DLR)
- heating and cooling demand (HDD/CDD)

It can also perform **land eligibility analyses**.

Rule: `build_renewable profiles`

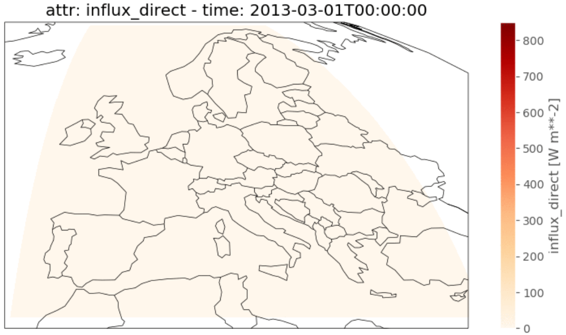


Time series for renewables

atlite: Convert weather data to energy systems data

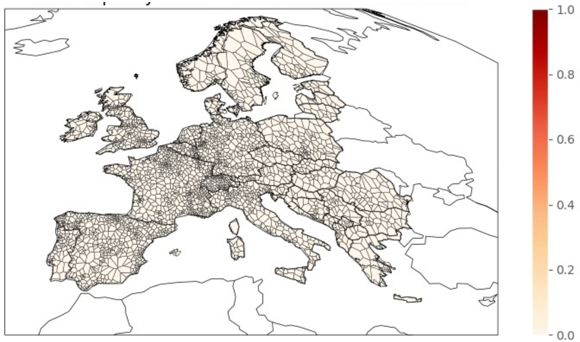
pypi v0.3.0 | conda-forge v0.3.0 | Tests passing | codecov 72% | docs passing | license MIT
REUSE compliant | JOSS 10.21105/joss.03294 | chat 52 online | stackoverflow pypsa questions 44

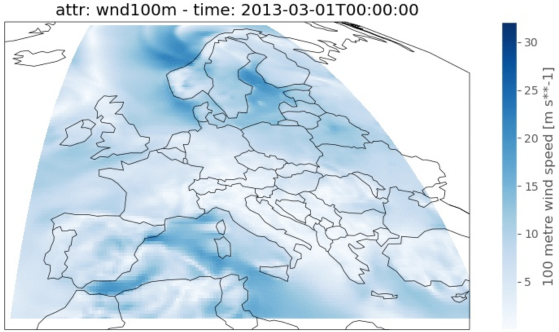
Historical meteorological weather data from ERA5 and SARAH-3
(up to 84 years, 30x30 km)



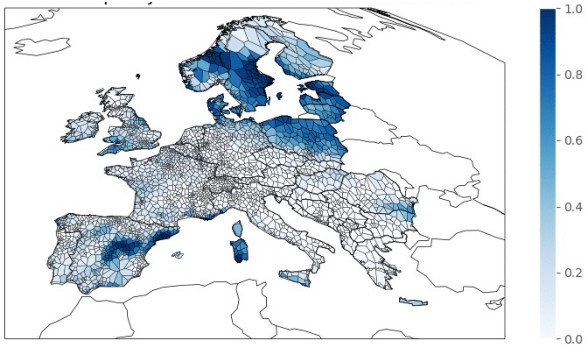
Solar panel models
- orientation
- material

Wind and solar capacity factors





Wind turbine models
- power curve
- surface roughness



Land availability for renewables

Example:
Onshore wind
in one clustered
region

atlite: Convert weather data to energy systems data

- CORINE / LUISA land cover
  - eligible land types
  - distance requirements
- NATURA / WDPA natural protection areas
- GEBCO bathymetry data
- Shipping lanes
- Distance to shore

Steps to building PyPSA-Eur electricity system

Retrieve onshore & offshore polygons for each country	`build_shapes`
Construct a base high-voltage network with buses, transformers, AC & DC lines with DLR & TYNDP	`base_network`, `build_transmission_projects`
Transform all transmission lines to 380kV, remove dead ends & cluster with k-means or hierarchical clustering	`simplify_network`, `cluster_network`
Determine eligible areas for utility-scale PV & onshore/offshore wind park development	`determine_availability_matrix`
Build renewable capacity factor profiles for each clustered region based on land availability	`build_renewable_profiles`, `build_hydro_profile`
Prepare existing renewables and fossil power plants	`build_powerplants`

Welcome to powerplantmatching's documentation!

https://globalenergymonitor.org/projects/global-integrated-power-tracker/tracker-map/

pypi v0.7.0 | conda-forge v0.7.0 | python >=3.9 | Tests failing | docs passing | pre-commit.ci passed | Ruff
license GPLV3+ | DOI 10.5281/zenodo.3358985 | stackoverflow pypsa questions 44

A toolset for cleaning, standardizing and combining multiple power plant databases.

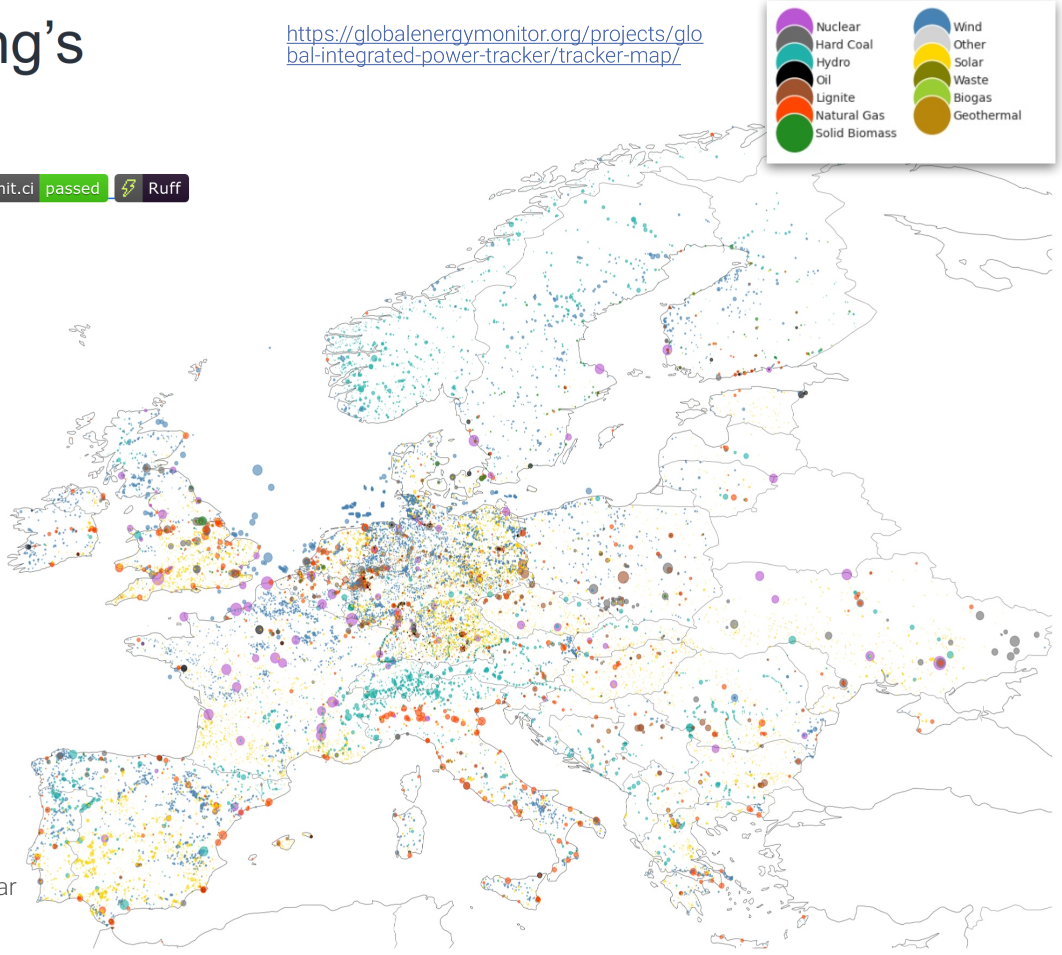
```
import powerplantmatching as pm
df = pm.powerplants(from_url=True)
df.query("DateIn > 2000")
```

Sources

- Global Energy Monitor (GEM)
- Open Power System Data (OPSD)
- Global Energy Observatory
- World Resources Institute
- Marktstammdatenregister (MaStR)
- CARMA
- ENTSO-E, BNetzA, UBA, IRENA
- JRC for hydro power plants

Attributes

- name
- fuel type
- technology
- country
- capacity
- commissioning year
- retirement year
- coordinates



github.com/pypsa/powerplantmatching

Steps to building PyPSA-Eur electricity system

Retrieve onshore & offshore polygons for each country	`build_shapes`
Construct a base high-voltage network with buses, transformers, AC & DC lines with DLR & TYNDP	`base_network`, `build_transmission_projects`
Transform all transmission lines to 380kV, remove dead ends & cluster with k-means or hierarchical clustering	`simplify_network`, `cluster_network`
Determine eligible areas for utility-scale PV & onshore/offshore wind park development	`determine_availability_matrix`
Build renewable capacity factor profiles for each clustered region based on land availability	`build_renewable_profiles`, `build_hydro_profile`
Prepare existing renewables and fossil power plants	`build_powerplants`
Add generation, storage and demand to the network with techno-economic assumptions on costs and efficiencies, …	`add_electricity`, `prepare_network`

Open database of techno-economic assumptions

- compiles **techno-economic assumptions** on energy system components
  - investment costs, FOM/VOM costs, efficiencies, lifetimes
  - for given years, e.g. 2020, 2030, 2040, 2050
  - from mixed sources, but prioritising **Danish Energy Agency** where available (and sensible)

Preview Code Blame 1097 lines (1097 loc) · 213 KB Raw

fischer-tropsch

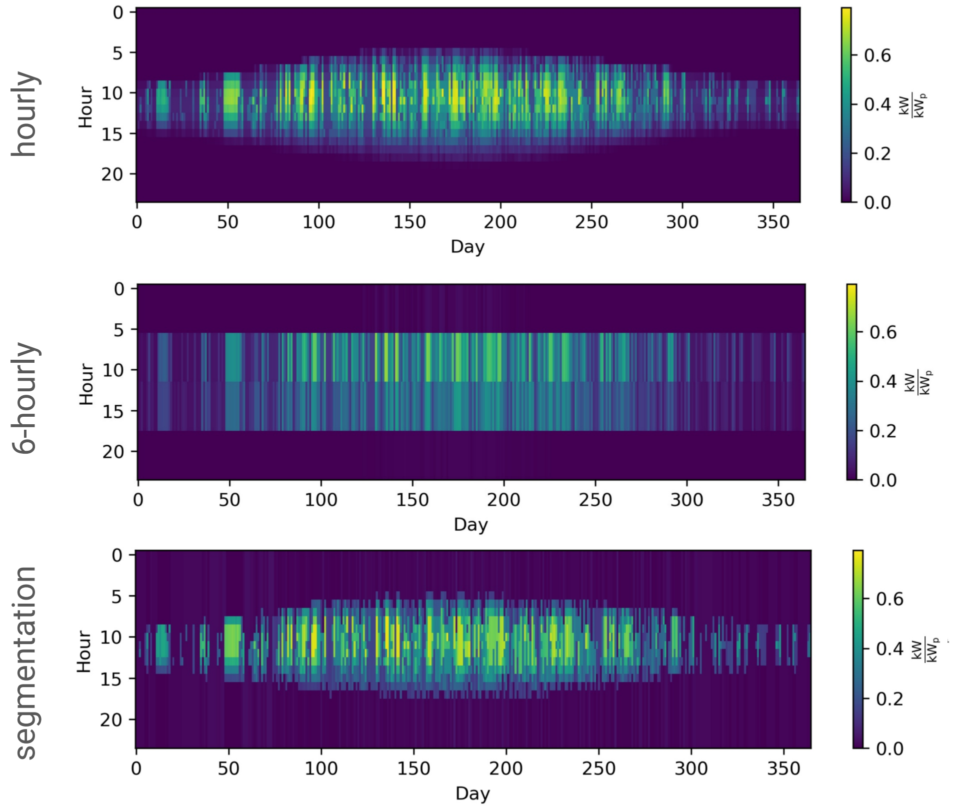
1	technology	parameter	value	unit	source	fu
217	Fischer-Tropsch	FOM	3.0	%/year	Agora Energiewende (2018): The Future Cost of Electricity-Based Synthetic Fuels (https://www.agora-energiewende.de/en/publications/the-future-cost-of-electricity-based-synthetic-fuels-1/), section 6.3.2.1.	
218	Fischer-Tropsch	VOM	4.4663	EUR/MWh_FT	Danish Energy Agency, data_sheets_for_renewable_fuels.xlsx	1(
219	Fischer-Tropsch	capture rate	0.9	per unit	Assumption based on doi:10.1016/j.biombioe.2015.01.006	
220	Fischer-Tropsch	carbondioxide-input	0.326	t_CO2/MWh_FT	DEA (2022): Technology Data for Renewable Fuels (https://ens.dk/en/our-services/projections-and-models/technology-data/technology-data-renewable-fuels), Hydrogen to Jet Fuel, Table 10 / pg. 267.	Ir
221	Fischer-Tropsch	efficiency	0.799	per unit	Agora Energiewende (2018): The Future Cost of Electricity-Based Synthetic Fuels (https://www.agora-energiewende.de/en/publications/the-future-cost-of-electricity-based-synthetic-fuels-1/), section 6.3.2.2.	
222	Fischer-Tropsch	electricity-input	0.007	MWh_el/MWh_FT	DEA (2022): Technology Data for Renewable Fuels (https://ens.dk/en/our-services/projections-and-models/technology-data/technology-data-renewable-fuels), Hydrogen to Jet Fuel, Table 10 / pg. 267.	0.
223	Fischer-Tropsch	hydrogen-input	1.421	MWh_H2/MWh_FT	DEA (2022): Technology Data for Renewable Fuels (https://ens.dk/en/our-services/projections-and-models/technology-data/technology-data-renewable-fuels), Hydrogen to Jet Fuel, Table 10 / pg. 267.	0.
224	Fischer-Tropsch	investment	703726.4462	EUR/MW_FT	Agora Energiewende (2018): The Future Cost of Electricity-Based Synthetic Fuels (https://www.agora-energiewende.de/en/publications/the-future-cost-of-electricity-based-synthetic-fuels-1/), table 8: "Reference scenario".	W
225	Fischer-Tropsch	lifetime	20.0	years	Danish Energy Agency, Technology Data for Renewable Fuels (04/2022), Data sheet "Methanol to Power".	
956	methanation	lifetime	20.0	years	Guesstimate.	B

https://github.com/PyPSA/technology-data/blob/master/outputs/costs_2030.csv

Temporal aggregation

Multiple options:

1. **averaging** of every Nth hour
2. **sampling** every Nth hour (e.g. 3-hourly)
3. Non-equidistant **segmentation** with pre-defined number of segments using the **tsam** Python library from **FZ Jülich**



JÜLICH Forschungszentrum

tsam - time series aggregation module

https://tsam.readthedocs.io/en/latest/newsDoc.html

Simplified workflow structure

Automated **downloads**

Process the data

Build the electricity system model

Extend the model by other sectors

(later)

Solve, summarize & visualize the results

linopy: Linear optimization with N-D labeled variables

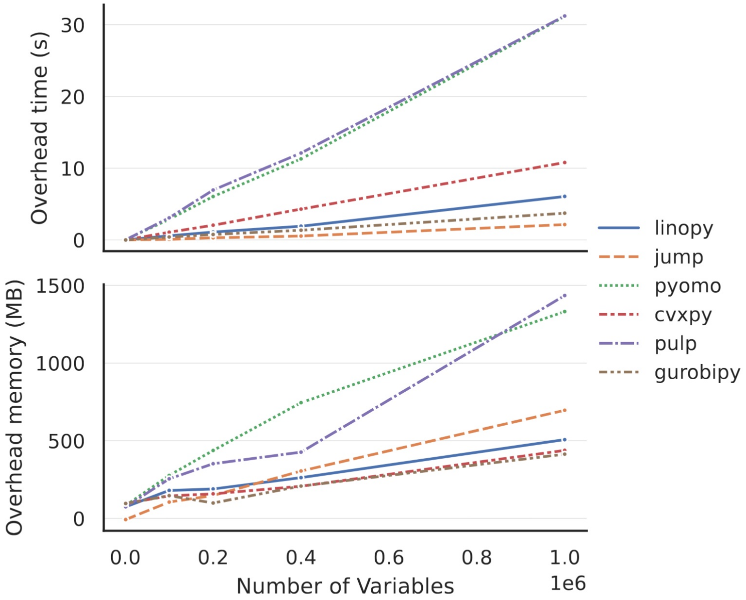
pypi v0.5.0 CI license MIT

Python library that facilitates **optimization** with real-world, large-scale data.

It supports:

- Linear (LP),
- Mixed-Integer (MILP),
- Quadratic programming (QP).

It has been developed to make linear programming in Python easy, highly-flexible and – most importantly – **highly performant**.



https://joss.theoj.org/papers/10.21105/joss.04823

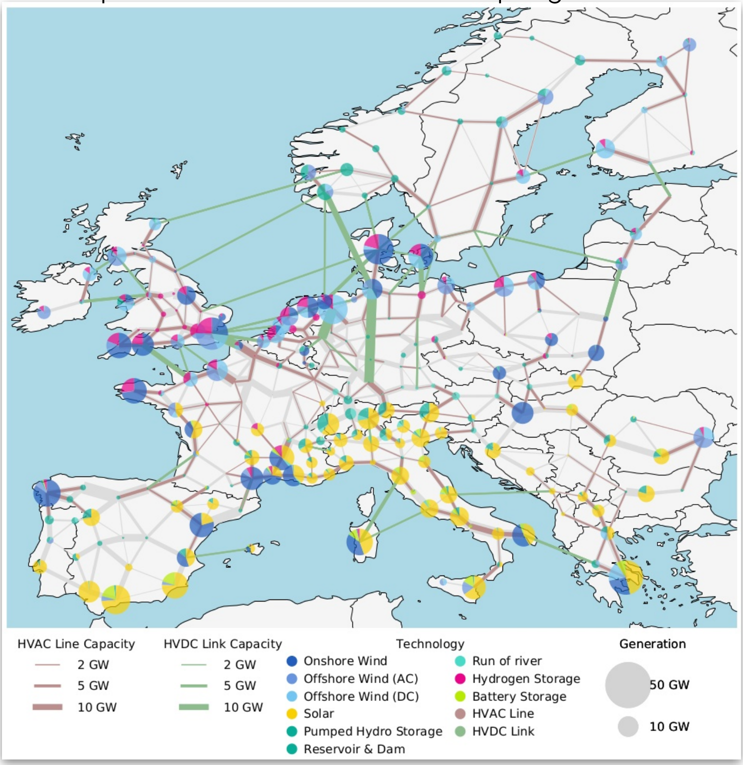
Solving and summarising networks

Hardware requirements:

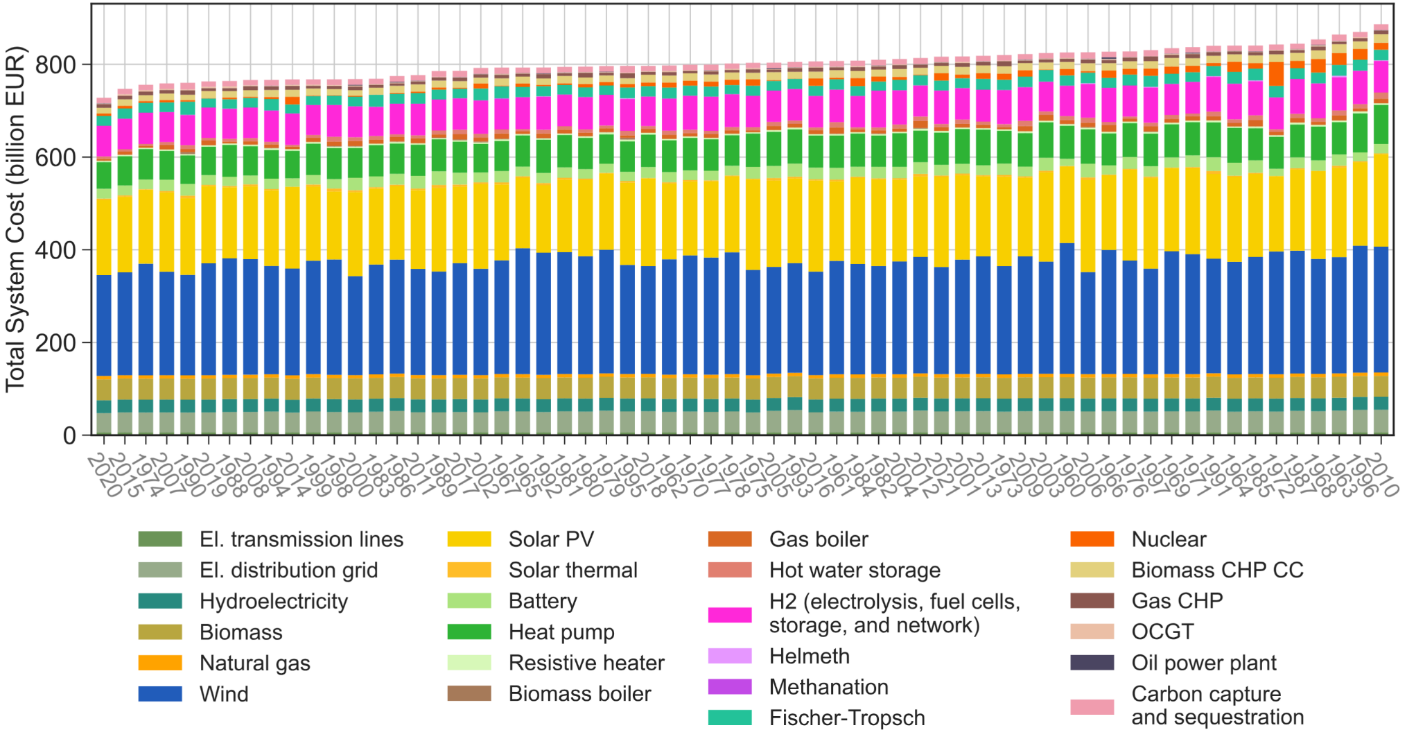
- Building the model **can run locally** on most modern laptops. Very simple models can run with HiGHS solver.
- But access to a **commercial solver** and a larger **cluster/workstation** is required for solving problems (~250 GB RAM per scenario if resolution is very high)!

There is a **statistics module in PyPSA** designed to help with analysing solved networks and several **figures/maps are created automatically**.

Example result without sector-coupling



PyPSA-Eur can be run on different **weather years**!



The years **2010**, **2013**, **2019** and **2023** are currently available "out of the box".

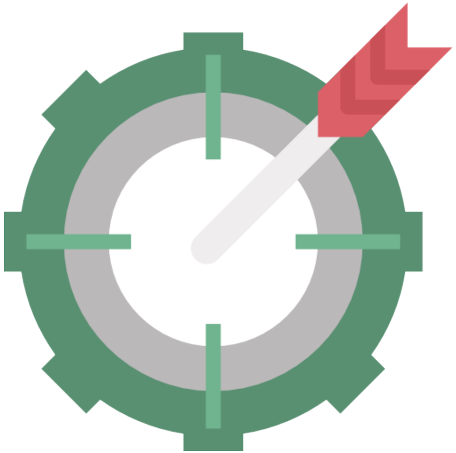
Other years **1940-2024** require a few more steps.

We are planning to expand the number of "plug-and-play" years.

https://www.nature.com/articles/s41467-024-54853-3

What is configurable?

electricity-only examples



config.yaml

- Select subset of countries and focus countries (e.g. only DE)
- Select weather year (1940 - 2024 for ERA5)
- Specify CO_2 constraint and gas usage limit
- Tweak spatial resolution (between 41 and >1000 nodes)
- Tweak temporal resolution (from hourly to N-hourly)
- Customize cost assumptions (e.g. 2020, 2030, 2050)
- Parametrize technologies (e.g. wind turbine type, panel orientation)
- Define land use eligibility criteria (e.g. distance requirements)
- Pick a solver (HiGHS, Gurobi, CPLEX, Xpress…)
- Choose between greenfield or brownfield expansion

Let’s look at this in more detail!

Icon made by surang from www.flaticon.com

Search Ctrl + K

Getting Started

- Introduction
- Installation
- Tutorial: Electricity-Only
- Tutorial: Sector-Coupled

Configuration

- Wildcards
- **Configuration**
- Foresight Options
- Techno-Economic Assumptions

Rules Overview

- Retrieving Data
- Building Electricity Networks
- Building Sector-Coupled Networks
- Solving Networks
- Plotting and Summaries

Configuration

PyPSA-Eur has several configuration options which are documented in this section and are collected in a `config/config.yaml` file. This file defines deviations from the default configuration (`config/config.default.yaml`); confer installation instructions at Handling Configuration Files.

Top-level configuration

"Private" refers to local, machine-specific settings or data meant for personal use, not to be shared. "Remote" indicates the address of a server used for data exchange, often for clusters and data pushing/pulling.

```
version: v2025.01.0
tutorial: false

logging:
  level: INFO
  format: '%(levelname)s:%(name)s:%(message)s'

private:
  keys:
    entsoe_api:

remote:
  ssh: ""
  path: ""
```

	Unit	Values	Description
version	–	0.x.x	Version of PyPSA-Eur. Descriptive only.
tutorial	bool	{true, false}	Switch to retrieve the tutorial data set instead of the full data set.
logging			

Contents

https://pypsa-eur.readthedocs.io/en/latest/configuration.html

Live Demo – Belgium / electricity-only / few days

Don't forget to activate your conda environment first!

Start with a dry-run:

$ snakemake solve_elec_networks --configfile config/test/config.electricity.yaml **-n**

The "-j1" flag tells snakemake to run one job at a time.

Then execute the same command "for real" by dropping "-n" flag:

$ snakemake -j1 solve_elec_networks --configfile config/test/config.electricity.yaml

To explore results, start a Jupyter notebook:

$ jupyter notebook

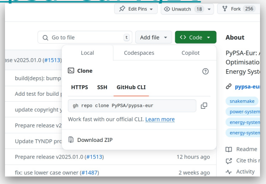
Practical Phase
(electricity-only)

1) Download the repository

Open a terminal / CMD and type:

```
$ cd ~/path/to/my/directory
$ git clone https://github.com/PyPSA/pypsa-eur.git
$ cd pypsa-eur
```

You can also download the repository as a ZIP by hand.



2) Install conda environment

Installation links:

- Anaconda (bigger download):
- Miniconda (recommended):

```
$ conda update conda
$ conda env create -f envs/environment.yaml
$ conda activate pypsa-eur
```

3) Run PyPSA-Eur tutorial with **snakemake**

Guide:
https://pypsa-eur.readthedocs.io/en/latest/tutorial.html

```
$ snakemake solve_elec_networks
  --configfile
  config/test/config.electricity.yaml
```

Users of Windows, add two lines to YAML:

```
run:
  use_shadow_directory: false
```

4) Explore PyPSA network in a Jupyter notebook

```
import pypsa
fn = "results/test-elec/networks/base_s_5_elec_.nc"
n = pypsa.Network(fn)
n.statistics()
n.plot()
```

Small exploratory configuration tasks

(electricity-only)

Go to https://pypsa-eur.readthedocs.io/en/latest/configuration.html and try to find out how to configure some of the settings for **electricity-only models** listed below:

1. Increase the maximum line loading from 70% to 100%.
2. Disable power transmission grid reinforcements.
3. Activate dynamic line rating with default settings.
4. Activate linearised transmission loss approximation.
5. Deactivate the estimation of existing renewable capacities.
6. Change the techno-economic assumptions to the year 2020.
7. Remove the option to build hydrogen or battery storage.

Simplified workflow structure

Automated **downloads**

Process the data

Build the electricity system model

Extend the model by other sectors

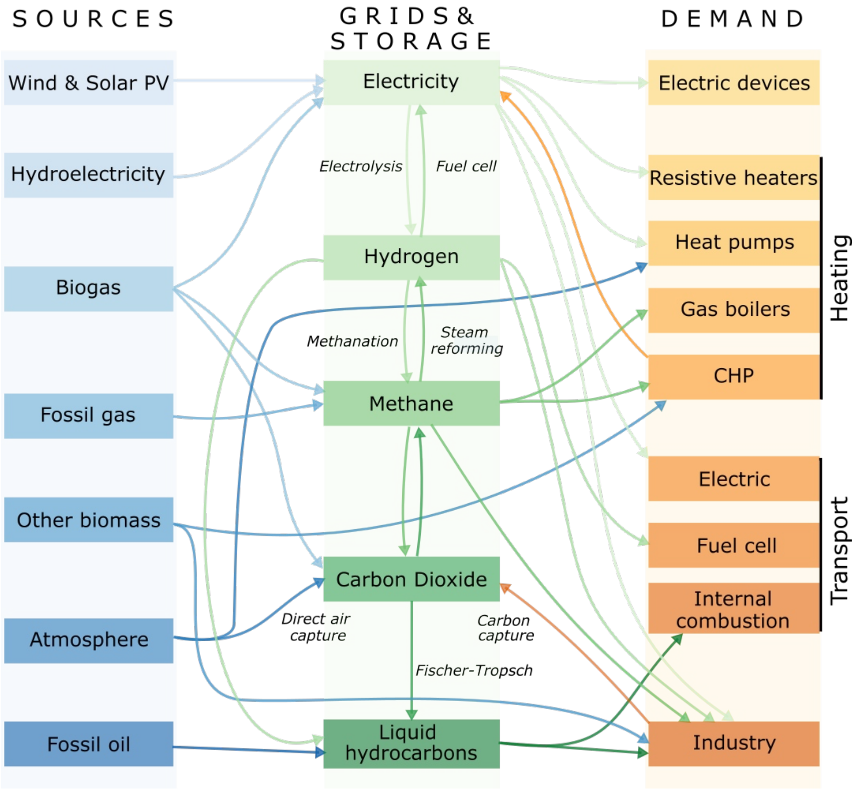
Solve, summarize & visualize the results

Coupling with other sectors

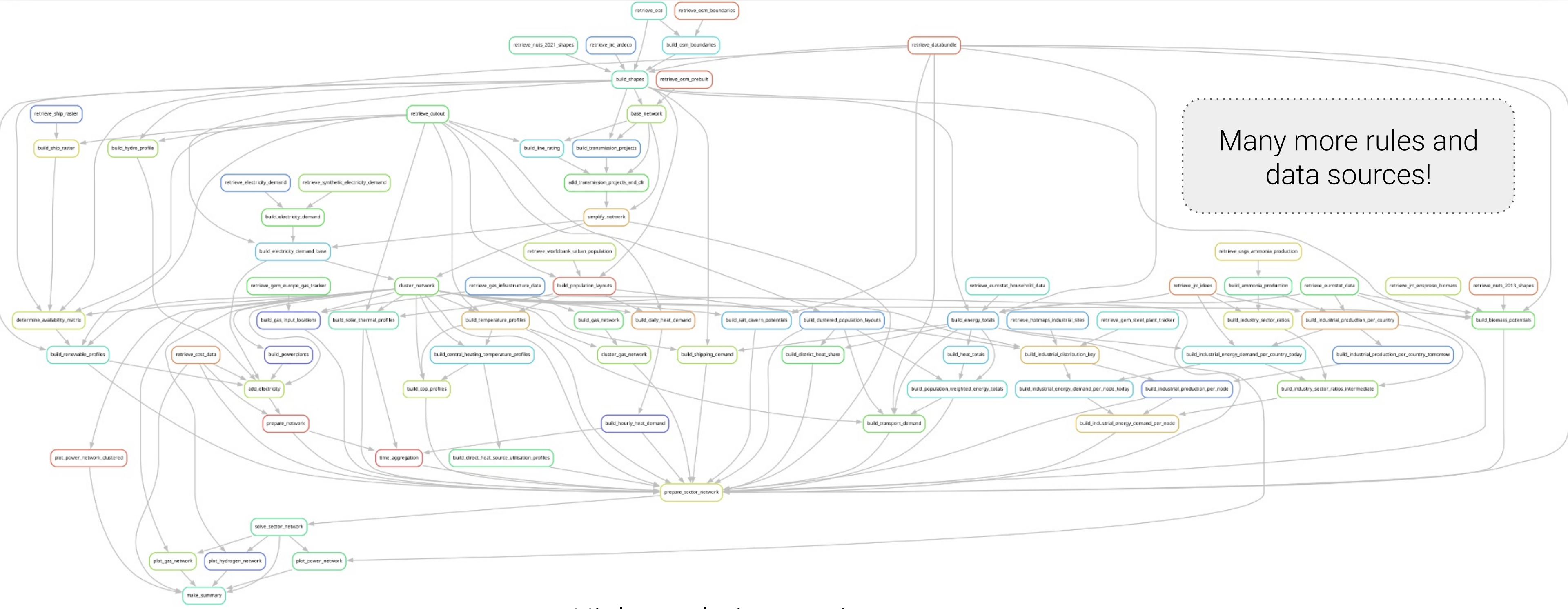
Need to decarbonise all sectors in Europe obeying spatial and temporal constraints.

- transport sector (EVs, shipping, aviation)
- heating sector (district heating, individual)
- industry sector (steel, chemicals, ammonia, …)
- industrial feedstocks
- biomass resources
- carbon management (CCUTS)
- hydrogen, CO_2 and gas networks
- pathway optimisation (myopic, perfect)

Boundaries between energy and material model blur.



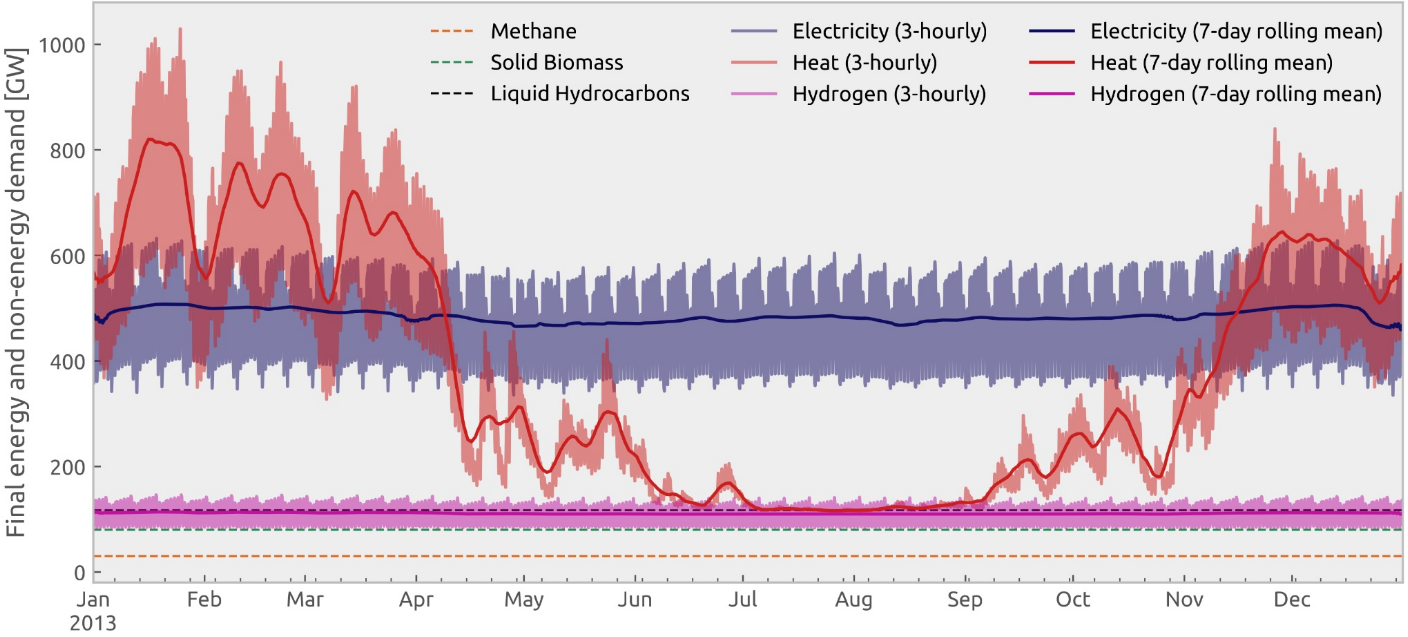
Extension by other sectors requires more data!



Many more rules and data sources!

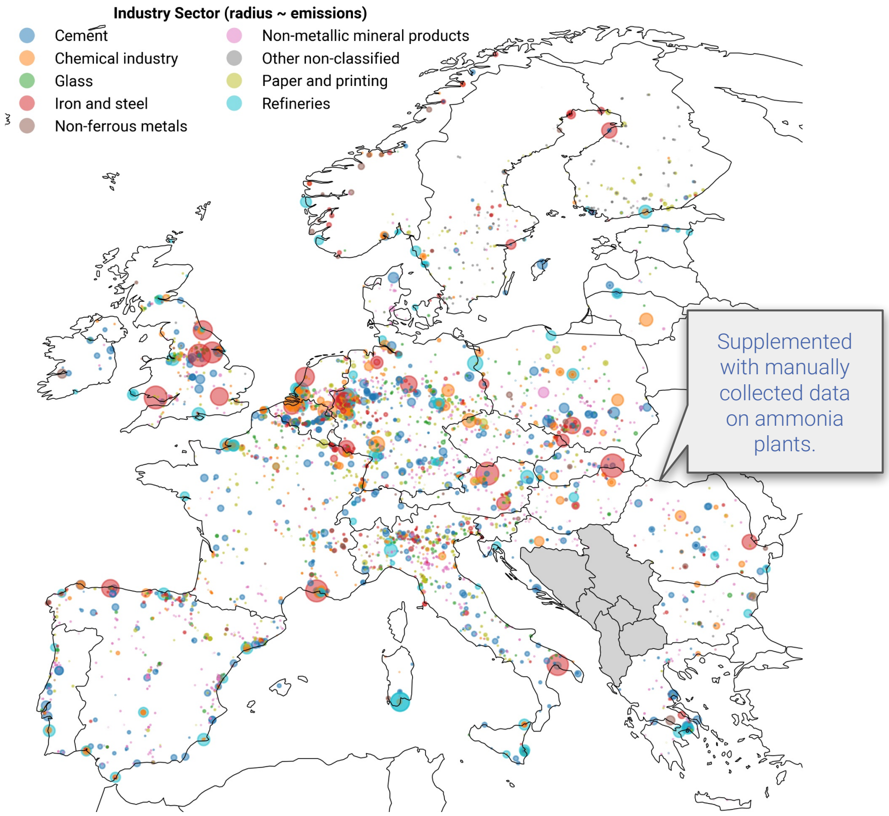
High-resolution version:
https://tubcloud.tu-berlin.de/s/E7tx3BagXsKXLre

Temporal distribution of energy demands



From a temporal perspective, the **seasonal variation of heat demand** adds a challenge – it can coincide periods of low wind and solar availability and varies from year to year.

Industry - Regionalisation based on Hotmaps

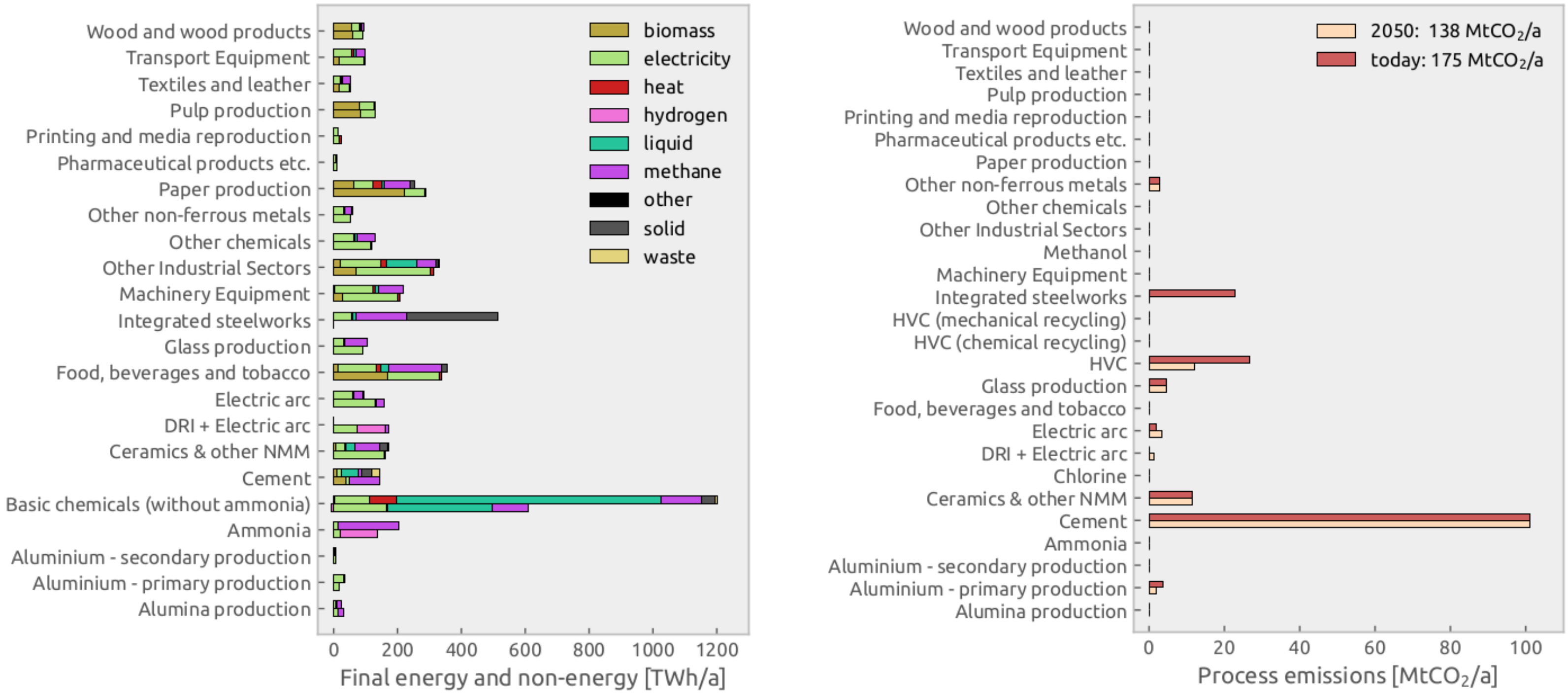


https://gitlab.com/hotmaps/industrial_sites/industrial_sites_Industrial_Database

Iron & Steel	Phase-out integrated steelworks; increased recycling; rest from H_2-DRI + EAF
Aluminium	Methane for high-enthalpy heat; increased recycling
Cement	Solid biomass; capture of CO_2 emissions
Ceramics	Electrification
Ammonia	Gray, blue, green hydrogen
Plastics	Synthetic naphtha; MtO/MtA, increased recycling
Other industry	Electrification; process heat from biomass
Shipping	Methanol, (oil), (liquid hydrogen), (LNG)
Aviation	Kerosene from Fischer-Tropsch or methanol

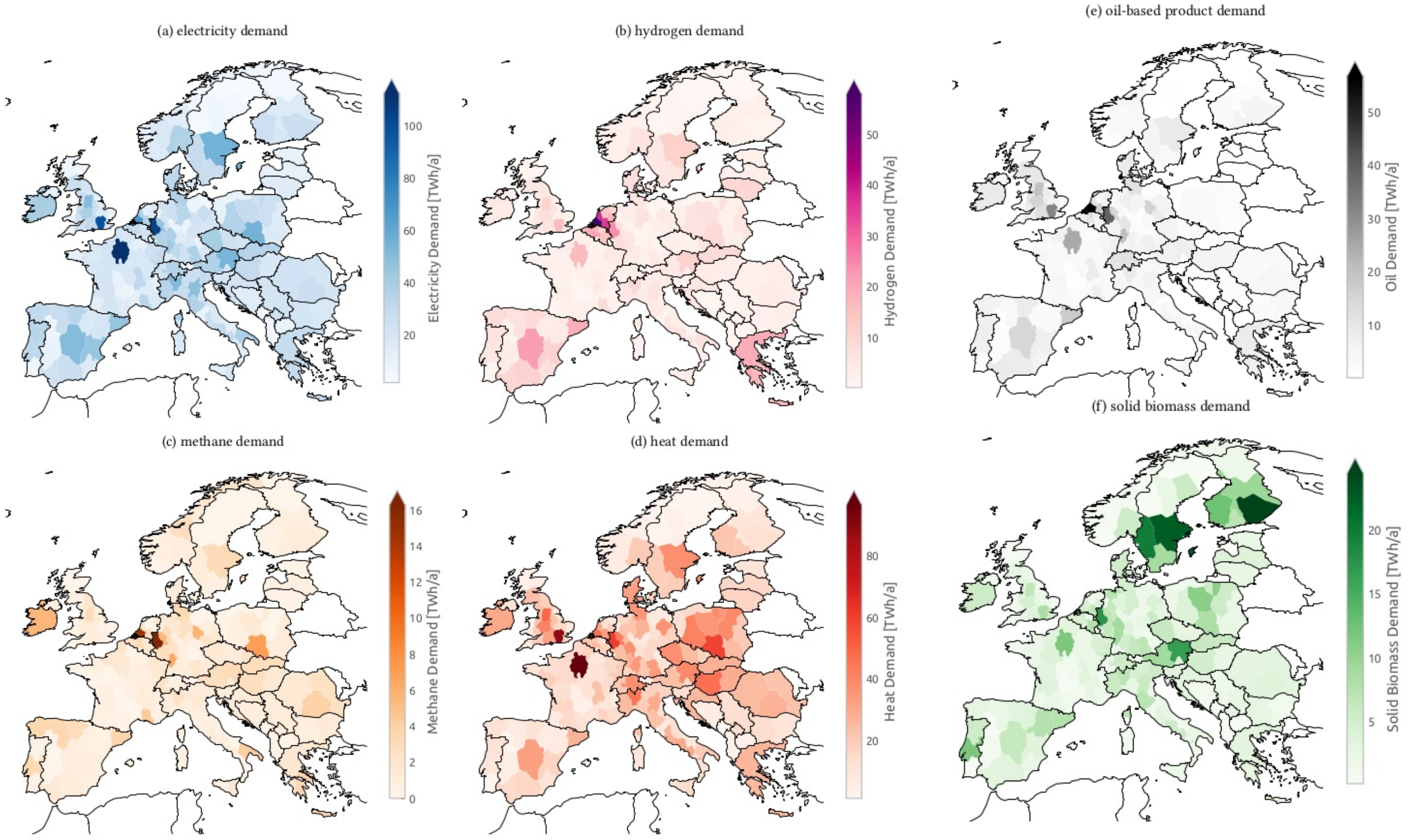
Modelling **industry relocation, high-temperature heat source & shipping fuels** endogenously is currently under development!

Industry - Fuel & process switching / process emissions



Currently, the most fuel & process switching in different industrial sectors is **exogenously configured** by the user. We're working to make these decisions **endogenous** to the model.

Spatial distribution of energy demands



Infrastructure - Gas network with H_2 retrofitting

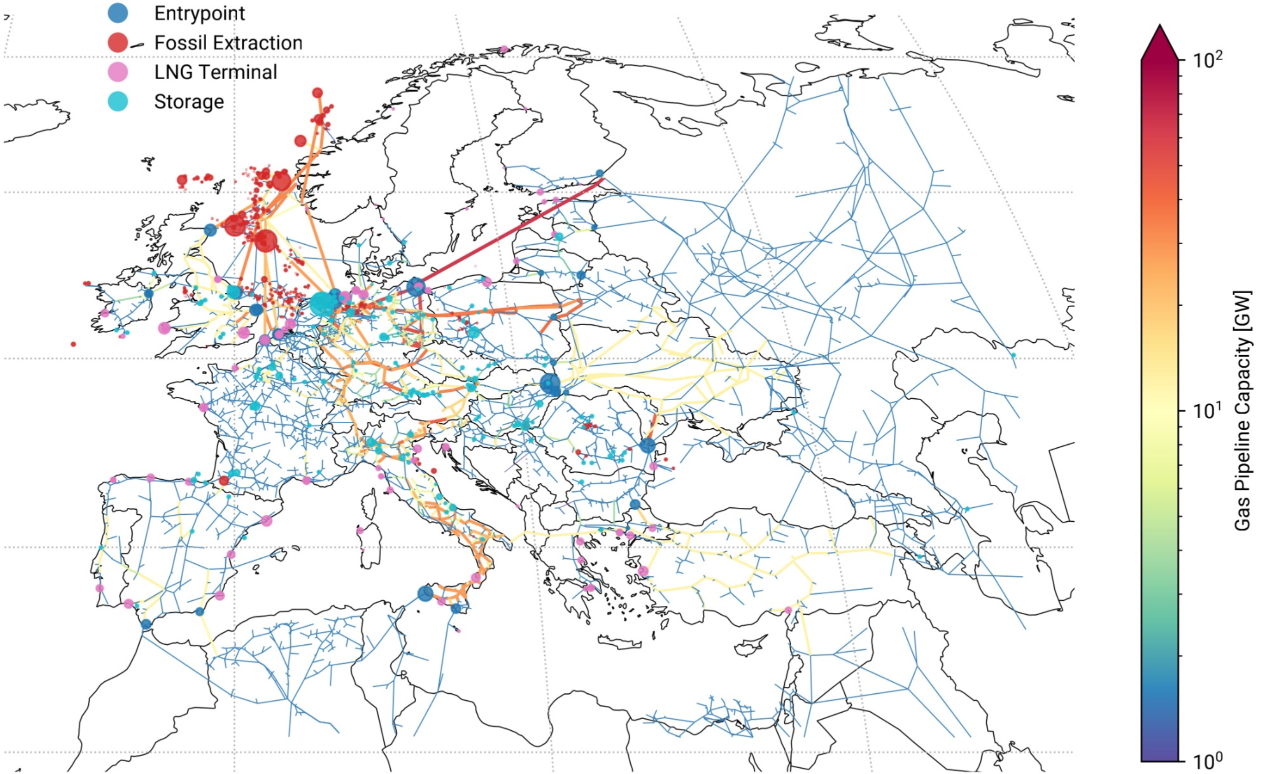
Compiled from open **SciGRID_gas** dataset.

Fossil gas enters at **LNG terminals** or **gas fields**.

Gas flow **physics** and **valve control** neglected 👉 transport model.

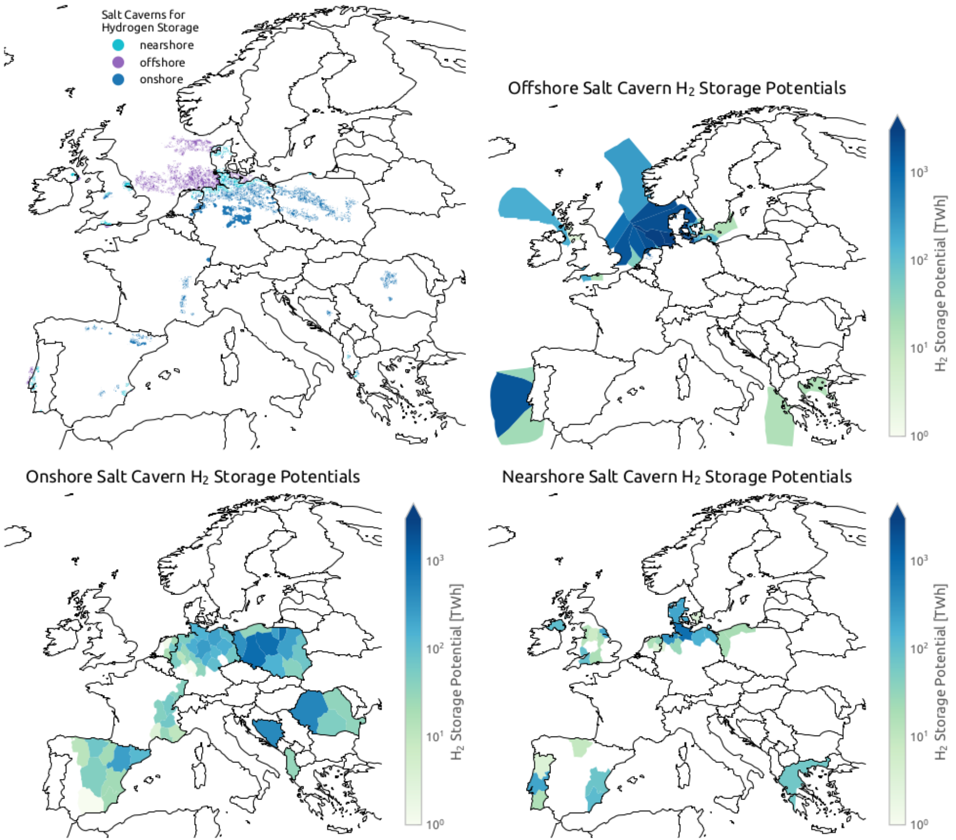
Electricity demand for **compression** and **leakage** configurable.

Pipelines can be **retrofitted** to H_2 with costs from EHB.



https://zenodo.org/records/4767098

Infrastructure - Hydrogen storage potentials

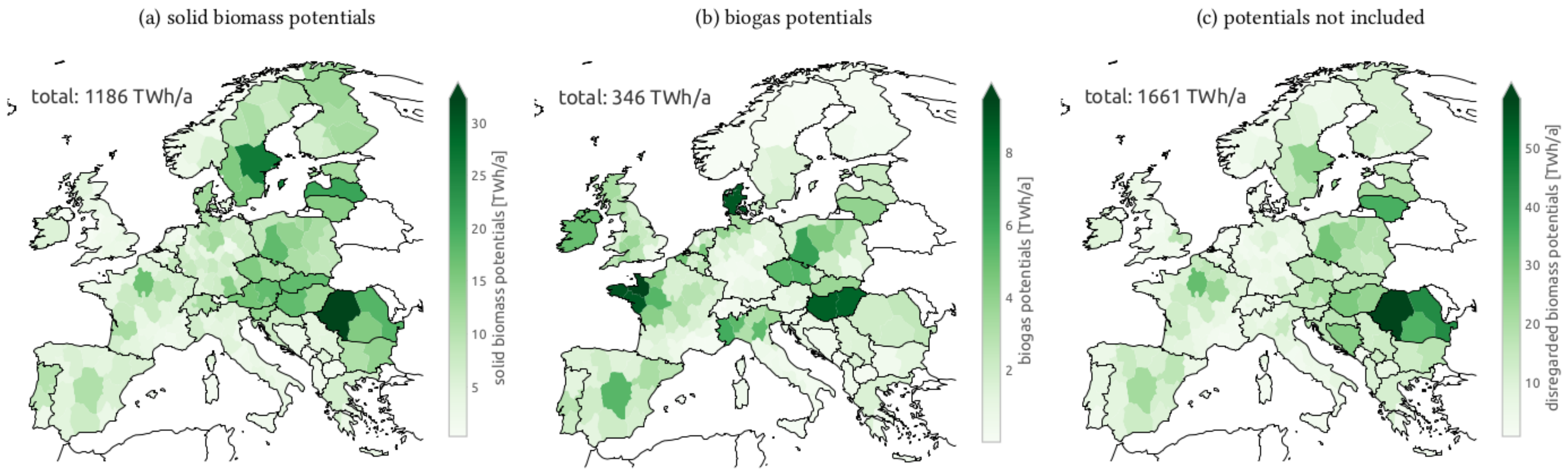


The regional distribution of **geological potential** to store hydrogen in **salt caverns** is considered.

The user can **configure** if onshore and/or offshore potentials can be used.

Dilara Gulcin Caglayan, Nikolaus Weber, Heidi U. Heinrichs, Jochen Linßen, Martin Robinius, Peter A. Kukla, Detlef Stolten, *Technical potential of salt caverns for hydrogen storage in Europe*, **International Journal of Hydrogen Energy**, Volume 45, Issue 11, 2020, 6793-6805, https://doi.org/10.1016/j.ijhydene.2019.12.161

Infrastructure - Biomass from JRC ENSPRESO



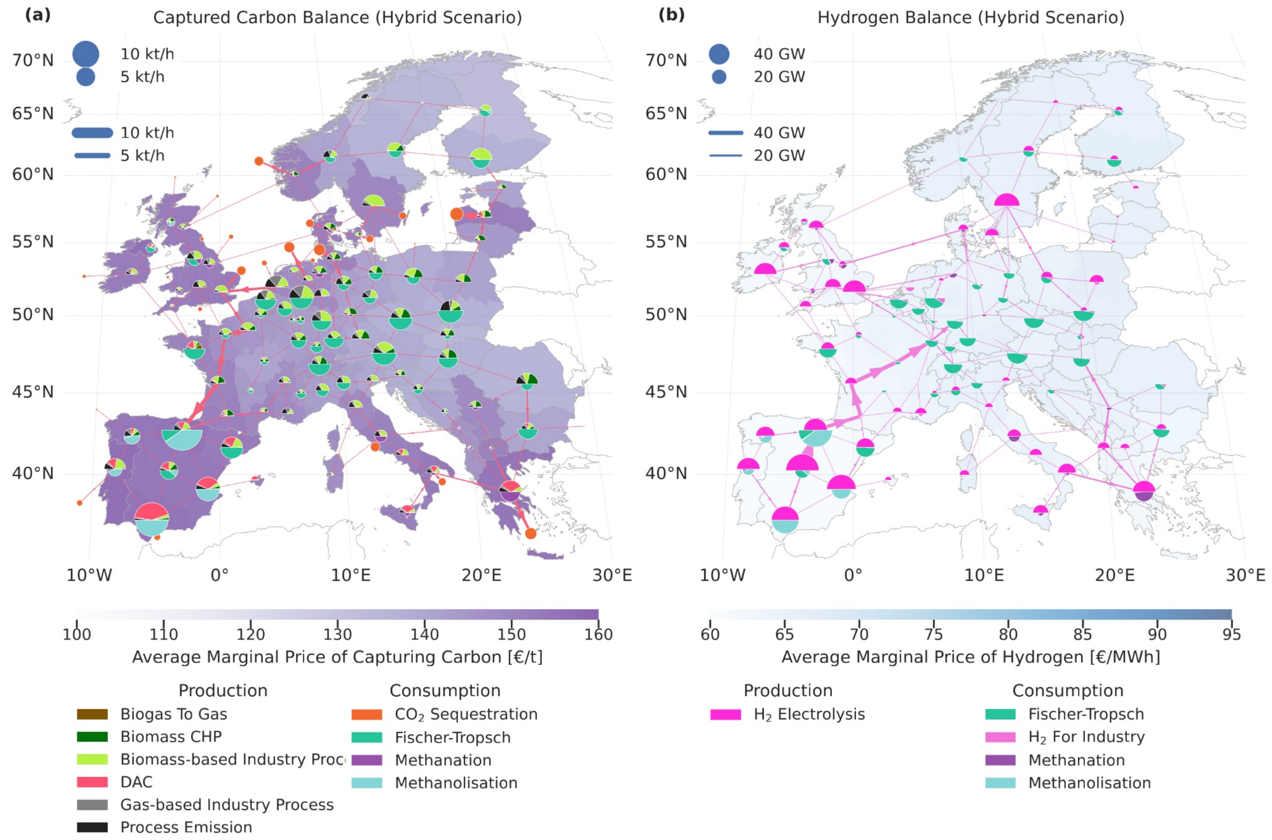
Biomass potentials are split between **solid biomass** and **biogas** (which can be, for instance, upgraded).

The user can configure low/medium/high potentials and what categories of biomass to consider (e.g. forest residues).

The default configuration only considers **residual biomass**, no energy crops.

https://data.jrc.ec.europa.eu/dataset/74ed5a04-7d74-4807-9eab-b94774309d9f

Infrastructure - Carbon management

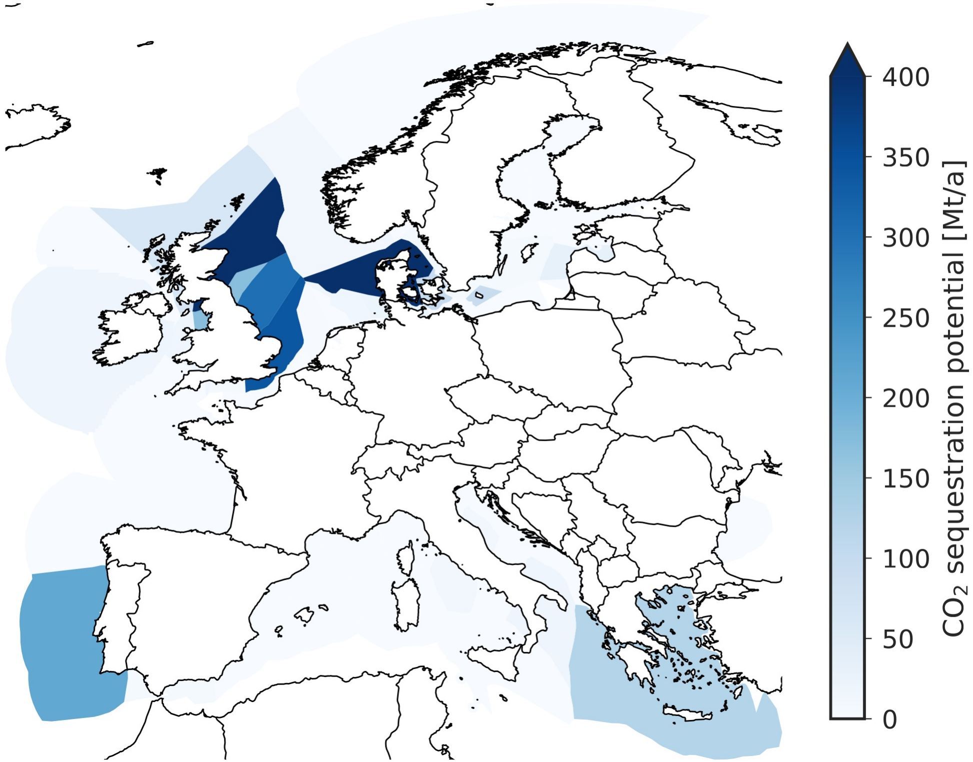


Built-in carbon flows:

- **Capture:** DAC, process emissions, fossil / biomass CHP
- **Transport:** CO_2 pipelines
- **Storage:** intermediate storage and long-term geological sequestration
- **Utilization:** for synthetic carbonaceous fuels

Infrastructure - Carbon sequestration potentials

Example: Offshore carbon sequestration potentials



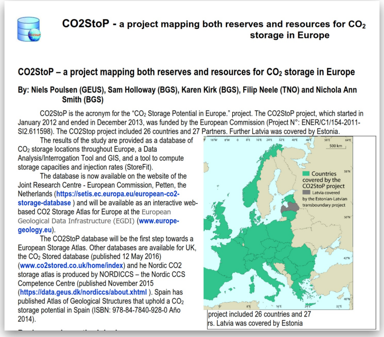
The user can **configure**

- onshore/offshore sequestration,
- gas fields/oil fields/aquifer, and
- low/medium/high potentials,

as well as a **total limit** on the annual sequestration, e.g. 250 Mt per year.

Data source:

https://energy.ec.europa.eu/publications/assessment-co2-storage-potential-europe-co2stop_en



CO2StoP - a project mapping both reserves and resources for CO_2 storage in Europe

CO2StoP – a project mapping both reserves and resources for CO_2 storage in Europe

By: Niels Poulsen (GEUS), Sam Holloway (BGS), Karen Kirk (BGS), Filip Neele (TNO) and Nichola Ann Smith (BGS)

CO2StoP is the acronym for the "CO_2 Storage Potential in Europe." project. The CO2StoP project, which started in January 2012 and ended in December 2013, was funded by the European Commission (Project N°: ENER/C1/154-2011-SI2.611598). The CO2Stop project included 26 countries and 27 Partners. Further Latvia was covered by Estonia.

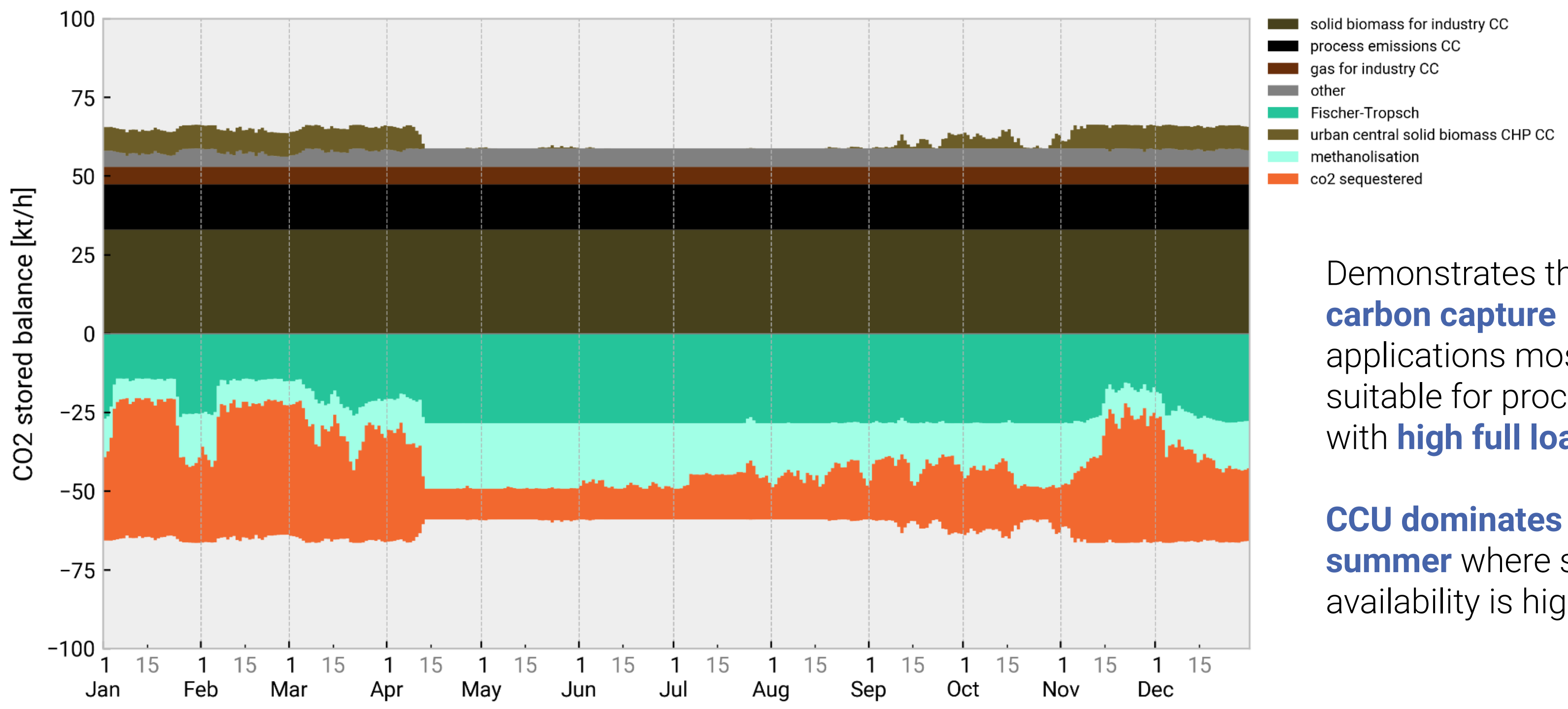
The results of the study are provided as a database of CO_2 storage locations throughout Europe, a Data Analysis/Interrogation Tool and GIS, and a tool to compute storage capacities and injection rates (StoreFit).

The database is now available on the website of the Joint Research Centre - European Commission, Petten, the Netherlands (https://setis.ec.europa.eu/european-co2-storage-database) and will be available as an interactive web-based CO2 Storage Atlas for Europe at the European Geological Data Infrastructure (EGDI) (www.europe-geology.eu).

The CO2StoP database will be the first step towards a European Storage Atlas. Other databases are available for UK, the CO_2 Stored database (published 12 May 2016) (www.co2stored.co.uk/home/index) and he Nordic CO2 storage atlas is produced by NORDICCS – the Nordic CCS Competence Centre (published November 2015 (https://data.geus.dk/nordiccs/about.xhtml). Spain has published Atlas of Geological Structures that uphold a CO_2 storage potential in Spain (ISBN: 978-84-7840-928-0 Año 2014).

project included 26 countries and 27
rs. Latvia was covered by Estonia

Examples - Carbon management on a time axis

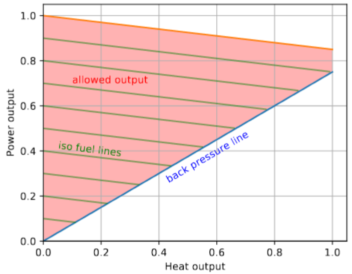


Demonstrates that **carbon capture** applications most suitable for processes with **high full load hours.**

CCU dominates over summer where solar availability is high.

Heating - Tech for individual & district heating

CHP feasible dispatch:



Decentral individual heating
can be supplied by:

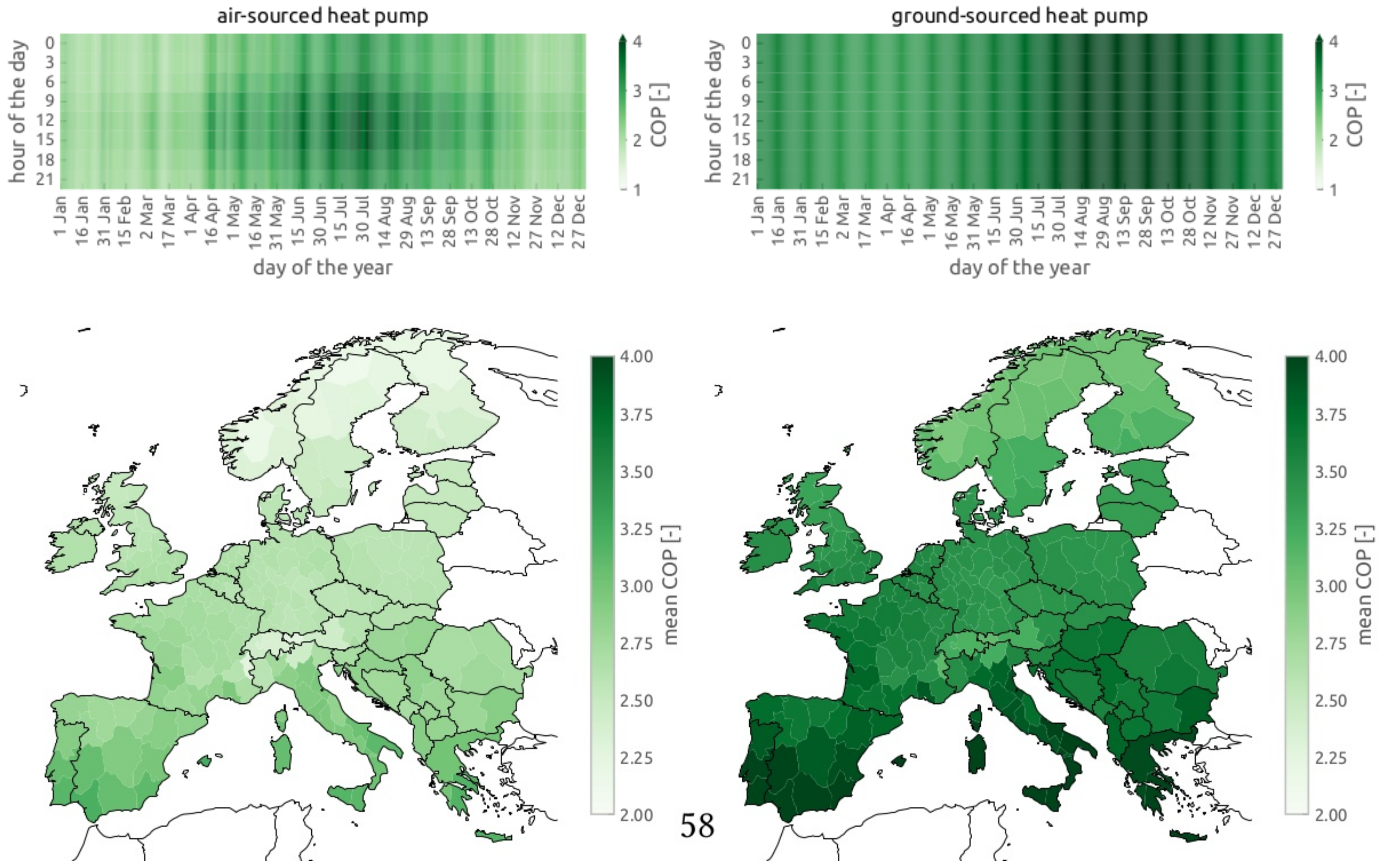
- air- or ground-sourced heat pumps
- resistive heaters
- gas / oil / biomass / hydrogen boilers
- solar thermal
- small water tanks

Building renovations can be co-optimized to reduce space heating demand.

District heating systems
can be supplied in urban areas by:

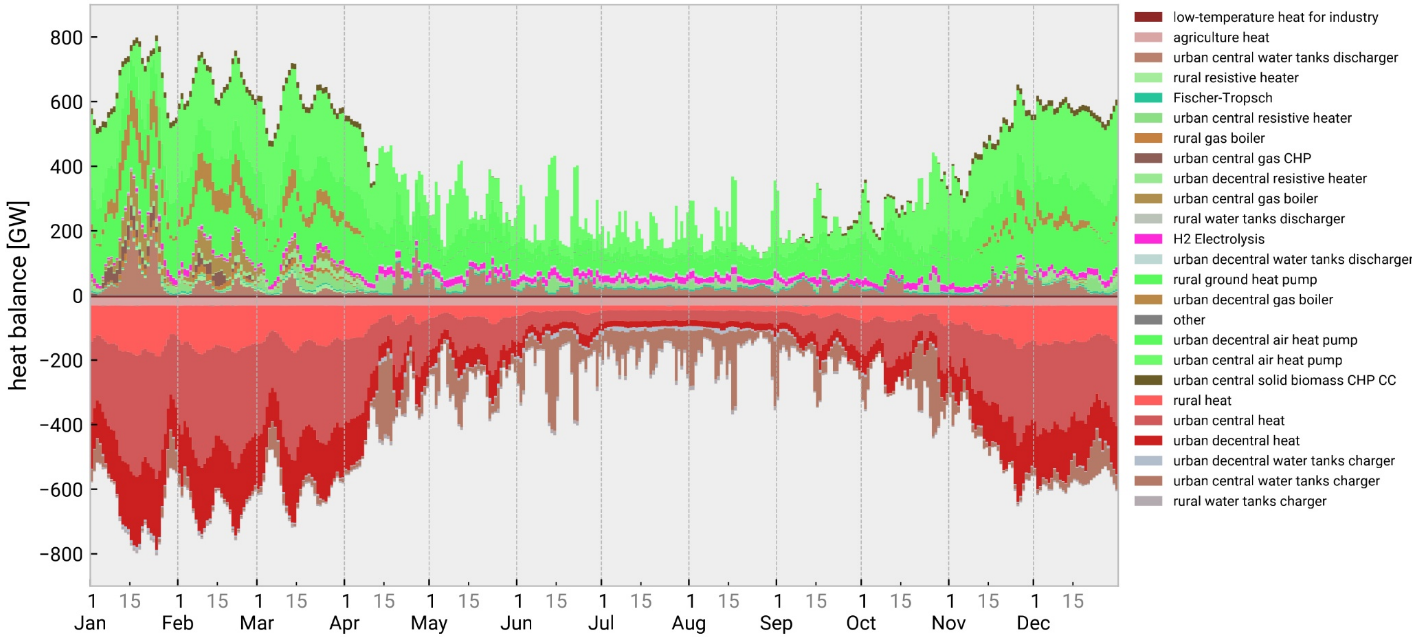
- air-sourced heat pumps
- resistive heaters
- gas / hydrogen / biomass / waste CHPs
- gas / oil / biomass / hydrogen boilers
- solar thermal
- long-duration hot water storage
- waste heat from industrial processes

Heating - Heat pumps as new variable supply tech



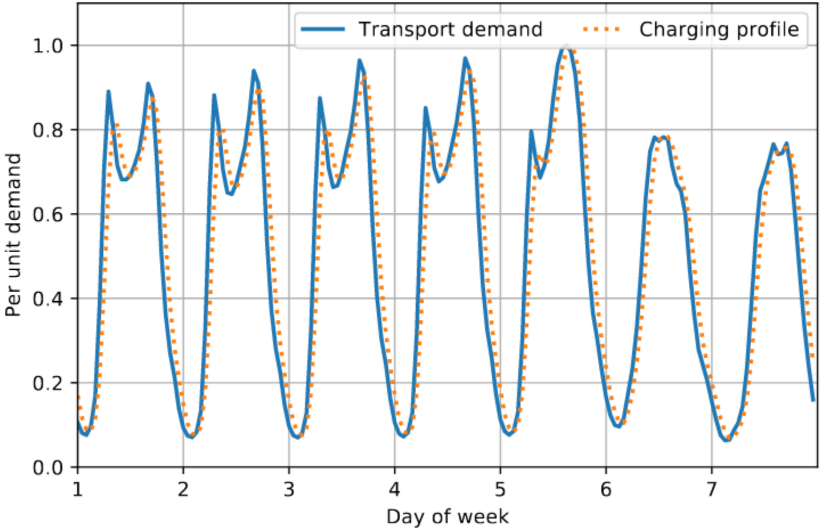
Geothermal heat sources have been integrated very recently!

Heating - Example daily heat system balance



There are difficult periods in winter with low wind and solar, high space heating demand and low air temperatures, which are bad for air-sourced heat pump performance. In this case **gas boilers** and **CHP plants** jump in as backup.

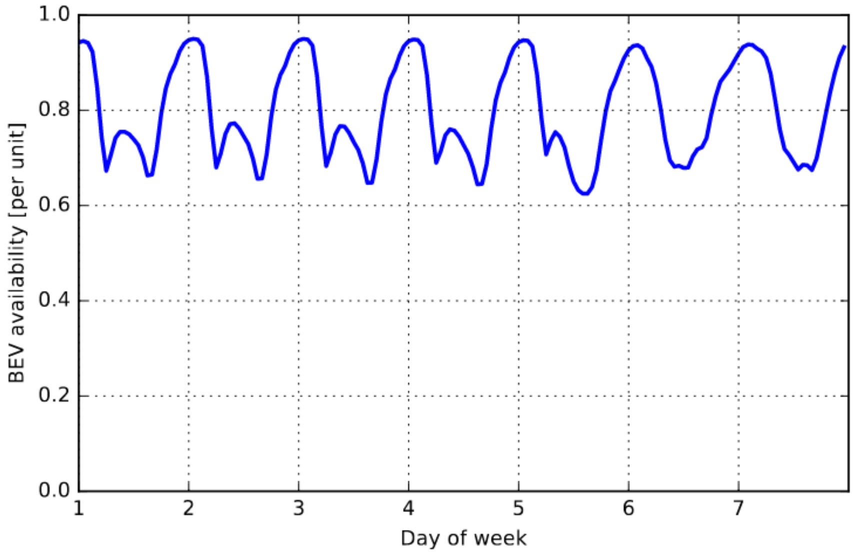
Transport - Electrification of land transport



Weekly profile for the transport demand based on statistics gathered by the German Federal Highway Research Institute (BASt).

- Road and rail transport is fully electrified (vehicle costs are not considered)
- Because of higher efficiency of electric motors, final energy consumption 3.5 times lower than today at 1100 TWh_{el}/a for Europe
- In model can replace Battery Electric Vehicles (BEVs) with Fuel Cell Electric Vehicles (FCEVs) consuming hydrogen. Advantage: hydrogen cheap to store. Disadvantage: efficiency of fuel cell only 60%, compared to 90% for battery discharging.

Transport - BEVs



Availability (i.e. fraction of vehicles plugged in) of Battery Electric Vehicles (BEV).

- Passenger cars to Battery Electric Vehicles (BEVs), 50 kWh battery available and 11 kW charging power
- Can participate in DSM and V2G, depending on scenario (state of charge returns to at least 75% every morning)
- All BEVs have time-dependent availability, averaging 80%, max 95% (at night)
- No changes in consumer behaviour assumed (e.g. car-sharing/pooling)
- BEVs are treated as exogenous (capital costs NOT included in calculation)

Technology choices - endogenous vs. exogenous

Exogenous assumptions (modeller chooses):

- energy services demand (e.g. heat)
- district heating shares
- energy carrier shares for road transport
- kerosene for aviation
- methanol for shipping
- electrification & recycling in industry
- steel production with DRI + EAF

Endogenous choices (model optimizes):

- change in electricity generation fleet
- transmission reinforcement
- capacities and locations of short and long-duration energy storage
- space and water heating technologies (including building renovations)
- all P2G/L/H/C
- supply of process heat for industry
- carbon capture (e.g. CHP, industry)

Supply, consumption and storage options by carrier

Electricity (115 regions)

Supply	Withdrawal
rooftop solar	industry electricity
utility-scale solar	residential electricity
onshore wind	services electricity
offshore wind (fixed-pole/floating, AC/DC-connected)	agriculture electricity
nuclear	air-sourced heat pump
hydro reservoirs	ground-sourced heat pump
pumped-hydro	resistive heater
run-of-river	electric vehicle charger
import by HVDC link	battery charger
gas CHP (w/wo CC)	pumped-hydro
biomass CHP (w/wo CC)	hydrogen pipeline (compression)
gas turbine (OCGT)	direct air capture
methanol turbine (OCGT)	Haber-Bosch
hydrogen turbine (OCGT)	electric arc furnace
hydrogen fuel cell CHP	direct iron reduction
battery discharger	distribution grid losses
vehicle-to-grid	transmission grid losses
	methanolisation
	electrolysis

Grids & Storage: distribution grid, transmission grid, battery storage, pumped-hydro storage, electric vehicles

Hydrogen (115 regions)

Supply	Withdrawal
import by pipeline	Fischer-Tropsch
import by ship	methanolisation
electrolysis	electrobiofuels
chlor-alkali electrolysis (exogenous)	direct iron reduction
steam methane reforming (w/wo CC)	Haber-Bosch
ammonia cracker	hydrogen turbine (OCGT)
	hydrogen fuel cell CHP
	methanol-to-kerosene
	Sabatier

Grids & Storage: new pipelines, retrofitted pipelines, storage in salt caverns, storage in steel tanks

Methane (not spatially resolved)

Supply	Withdrawal
import by ship	gas for high-T industry heat (w/wo CC)
fossil gas	steam methane reforming (w/wo CC)
biogas upgrading (w/wo CC)	gas boiler (rural/urban)
Sabatier	gas CHP
	gas turbine (OCGT)

Storage: hydrocarbon storage

Liquid Hydrocarbons (not spatially resolved)

Supply	Withdrawal
import by ship	kerosene for aviation
fossil oil refining	naphtha for industry
Fischer-Tropsch	diesel for agriculture
electrobiofuels	

Storage: hydrocarbon storage

Methanol (not spatially resolved)

Supply	Withdrawal
import by ship	methanol turbine (OCGT)
methanolisation	methanol for shipping
	methanol for industry
	methanol-to-kerosene

Storage: hydrocarbon storage

Ammonia (not spatially resolved)

Supply	Withdrawal
import by ship	ammonia cracker
Haber-Bosch	ammonia for fertilizer

Storage: ammonia tank

Supply, consumption and storage options by carrier

Heat (115 regions)

Supply	Withdrawal
air-sourced heat pump	residential heat
ground-sourced heat pump (only rural)	services heat
resistive heater	agriculture heat
gas boiler	low-T industry heat
biomass boiler	direct air capture
solar thermal	water tank charger
water tank discharger	
biomass CHP (w/wo CC, only DH)	
gas CHP (w/wo CC, only DH)	
hydrogen fuel cell CHP (only DH)	
electrolysis (only DH)	
Haber-Bosch (only DH)	
Sabatier (only DH)	
Fischer-Tropsch (only DH)	
methanolisation (only DH)	

Storage: long-duration thermal storage (only DH); hot water tank

CO2 atmosphere (not spatially resolved)

Supply	Withdrawal
kerosene for aviation	solid biomass for industry (w CC)
diesel for agriculture	solid biomass CHP (w CC)
methanol for shipping	biogas upgrading (w CC)
methanol for industry	direct air capture
naphtha for industry	electrobiofuels
gas boiler	
gas CHP (w/wo CC)	
gas turbine (OCGT)	
methanol turbine (OCGT)	
process emissions (w/wo CC)	
fossil oil refining	
gas for high-T industry heat (w/wo CC)	
steam methane reforming (w/wo CC)	

CO2 commodity (not spatially resolved)

Supply	Withdrawal
direct air capture	Fischer-Tropsch
biogas upgrading (w CC)	methanolisation
gas CHP (w CC)	sequestration
biomass CHP (w CC)	Sabatier
steam methane reforming (w CC)	
process emissions (w CC)	
solid biomass for industry (w CC)	
gas for high-T industry heat (w CC)	

Storage: intermediate storage in steel tank; long-term geological sequestration

Myopic pathway optimization

- Provide exogenous CO_2 emission **reduction path**.
- Optimise **start network** for e.g. 2025, starting with existing energy infrastructure.
- Take results from **2025 as input** for 2030 infrastructure optimisation, take 2030 results for next iteration, etc.
- The choice of **investment years** is arbitrary.
- **Perfect foresight pathway planning** is currently experimental (i.e. endogenous CO_2 budget).

Running many different scenarios with alternative configurations is straightforward and scalable in **snakemake!**

Live Demo – very similar to electricity-only case

Start with a dry-run:

```
$ snakemake all --configfile config/test/config.overnight.yaml -n
```

Then execute the same command “for real” by dropping “-n” flag:

```
$ snakemake all --configfile config/test/config.overnight.yaml
```

And for myopic pathway optimisation:

```
$ snakemake all --configfile config/test/config.myopic.yaml
```

To explore results, start a Jupyter notebook:

```
$ jupyter notebook
```

Practical Phase
(sector-coupled)

1) Run PyPSA-Eur sector-coupling tutorial with **snakemake**

   Guide:
   https://pypsa-eur.readthedocs.io/en/latest/tutorial_sector.html

   snakemake all --configfile config/test/config.overnight.yaml

2) Explore CSV files and images in **results** directory.

Users of Windows, add two lines to YAML:

```
run:
  use_shadow_directory: false
```

Small exploratory configuration tasks

(sector-coupled)

Go to https://pypsa-eur.readthedocs.io/en/latest/configuration.html and try to find out how to configure some of the settings for **sector-coupled models** listed below:

1. Disable vehicle-to-grid discharging.
2. Disable methanation as technology option.
3. Increase the carbon sequestration potential to 500 Mt/a.
4. Allow hydrogen underground storage also onshore.
5. Reduce the primary production of plastics by increasing recycling rates.
6. Change the settings of all transmission so that they are lossless.
7. Disable the use of PtX waste heat.

Scenario management

PyPSA-Eur has integrated & scalable scenario management!

config/config.yaml

```
run:
  name: all
  scenarios:
    enable: true

scenario:
  clusters: [90]

sector:
  H2_network: true
  gas_network: true
  H2_retrofit: true

electricity:
  transmission_limit:
vopt
```

With these two files configured, run:

```
$ snakemake all -n
```

and

```
$ snakemake all
```

config/scenarios.yaml

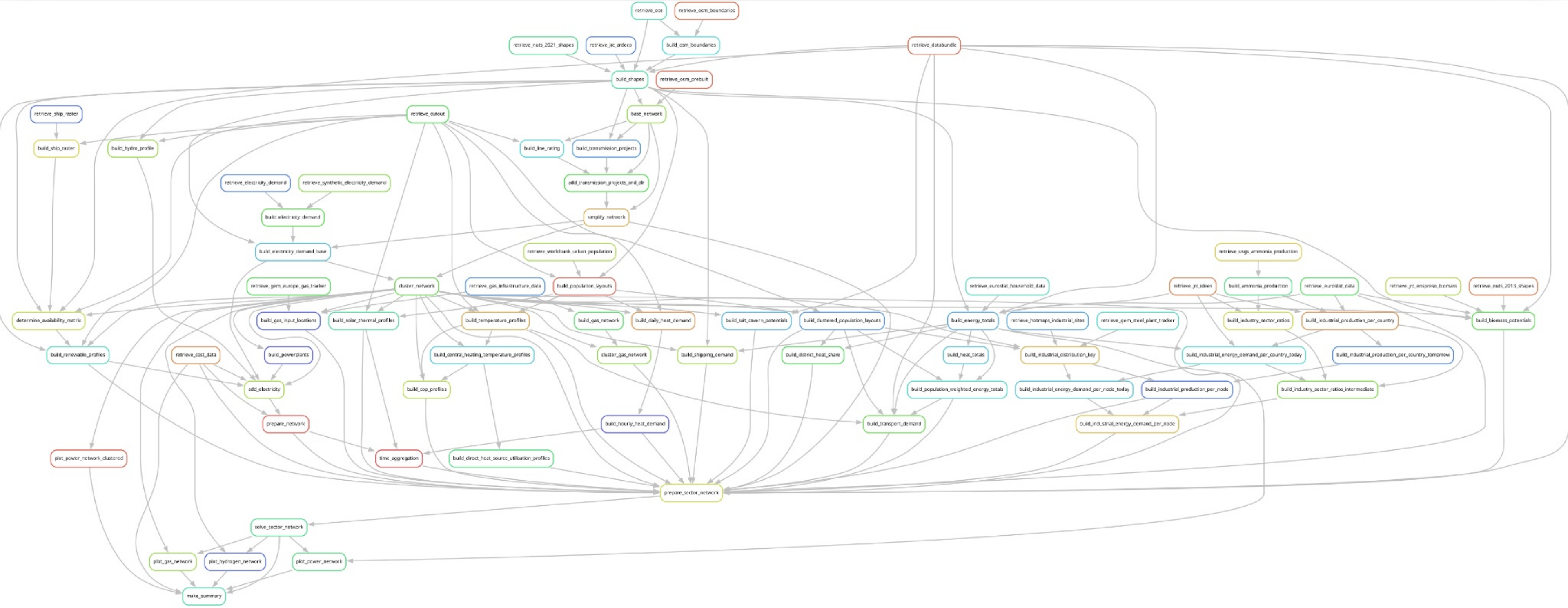
```
no-h2-network:
  sector:
    H2_network: false

no-grid-expansion:
  electricity:
    transmission: v1.0

no-to-both:
  sector:
    H2_network: false
  electricity:
            transmission:
v1.0

yes-to-both:
  sector:
            H2_network:
true
  electricity:
            transmission:
vopt
```

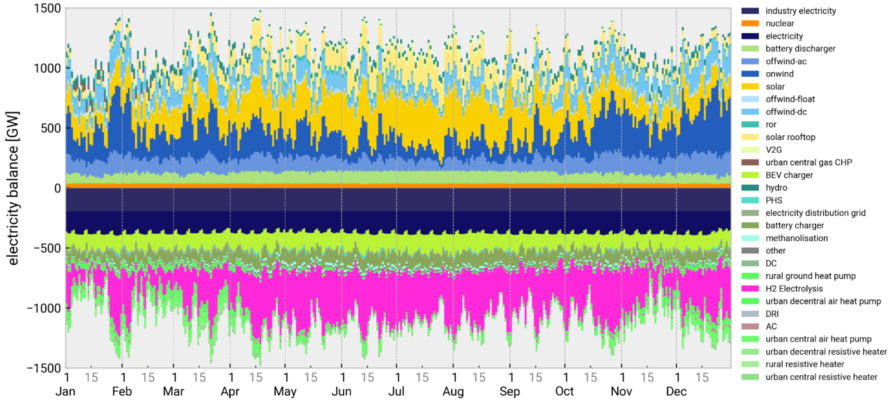
Closing remark – There is much more to explore!



Additional Resources

Documentation
https://pypsa-eur.readthedocs.io/

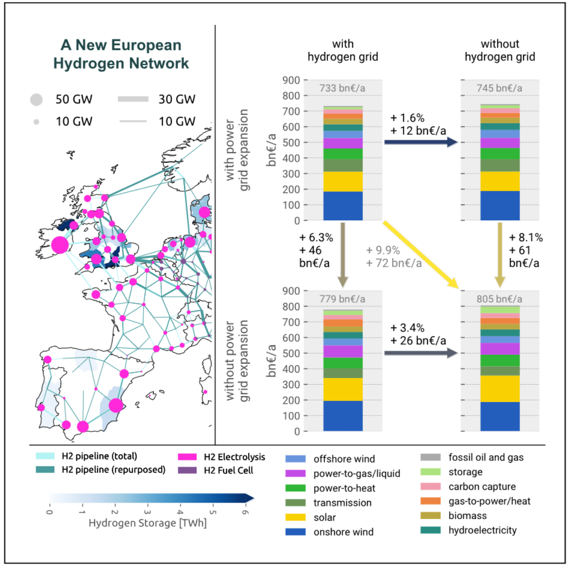
Supplementary Material
https://www.cell.com/joule/pdfExtended/S2542-4351(23)00266-0



Joule
CellPress OPEN ACCESS

Article

The potential role of a hydrogen network in Europe



Fabian Neumann, Elisabeth Zeyen, Marta Victoria, Tom Brown

f.neumann@tu-berlin.de

Highlights

Examination of the cost benefit of a European hydrogen network in net-zero emission scenarios

H_2 network reduces system costs by up to 3.4%, highest without power grid expansion

Between 64% and 69% of the hydrogen network uses retrofitted gas network pipelines

Power grid expansion saves more than hydrogen network, but strongest savings with both

We examine the interplay between a continent-wide hydrogen network and electricity grid expansion in Europe to help balance variations in wind and solar energy supply. By adapting existing natural gas pipelines for hydrogen transport, energy system costs can be reduced, especially when power grid reinforcements are not possible. Both types of transmission infrastructure offer cost-effective options for achieving a European energy system with net-zero CO_2 emissions. However, with a 10% cost increase, it is possible to build neither.