Tackling challenges in energy system research with HPC

Thomas Breuer[1], Karl-Kiên Cao[2], Manuel Wetzel[2], Ulrich Frey[2], Shima Sasanpour[2], Jan Buschmann[2], Kai von Krbek[2], Aileen Böhme[3], Charlie Vanaret[4,5]

Energy Systems Analysis Needs HPC

Background: The status quo

Studying the future is inherently subject to large uncertainties. The state of the art in energy system research is limited by the following three key challenges:

1. These uncertainties are tackled with ensemble modeling of a **small subset of all possible scenarios**. This has proven to be inadequate as the models are highly sensitive to certain input data
2. The widely-used commercial solvers show poor scalability and are **limited to single shared-memory compute nodes**. Thus, models are defined with a lower temporal and spatial resolution and a lower technological diversity than necessary to ensure applicability for real world policy support.
3. **Specialized models** tend to investigate only certain aspects, which do not cover all parts of future energy system pathways

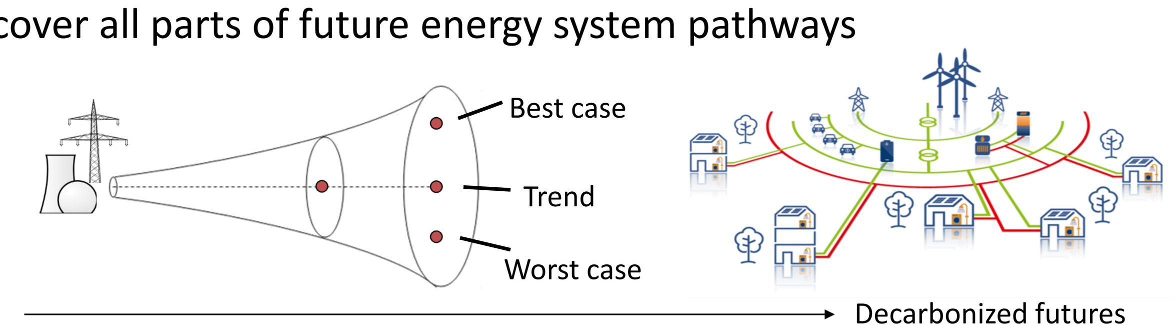

Fig. 1: The established scenario funnel for decarbonization pathways of energy systems

Objective: The theoretical best practice

We have opted to fully inspect the conceivable parameter space for the first time by using a hitherto unattained number of model-based energy scenarios. To overcome the above-mentioned limitation we use:

1. **Automated parameter sampling** based on a broad literature review
2. A self-developed **distributed memory solver**, called PIPS-IPM++[6]
3. **Coupling** of different **model** types and definition of relevant **indicators**

Efficiently leveraging the capability of HPC by combining those approaches in an **automated workflow** could be a game changer for the energy-system analysis community.

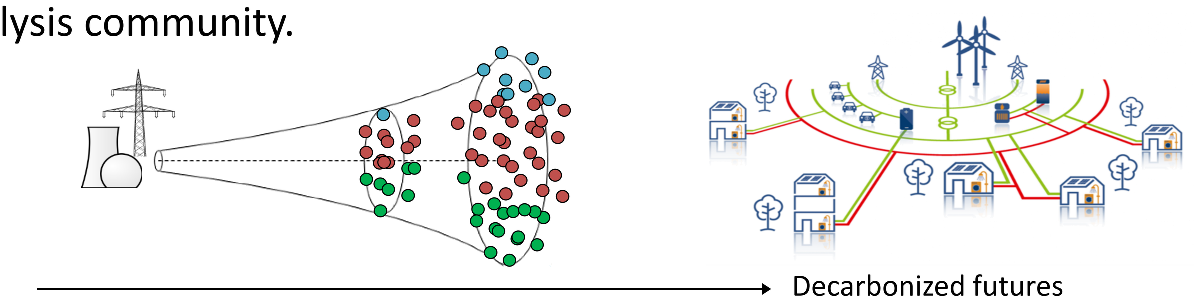

Fig. 2: The scenario funnel for decarbonization pathways of energy systems in UNSEEN

Acknowledgement and collaboration

The authors gratefully acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for funding this project by providing computing time through the John von Neumann Institute for Computing (NIC) on the GCS Supercomputer JUWELS at Jülich Supercomputing Centre (JSC).

UNSEEN[12] (grant number FKZ 03EI1004A; duration: 10/2019-03/2023) is an interdisciplinary consortium of researchers and practitioners from system analysis, mathematics, and high performance computing.

1 JÜLICH Forschungszentrum · 2 DLR Deutsches Zentrum für Luft- und Raumfahrt German Aerospace Center · 3 GAMS · 4 TU Berlin · 5 ZIB

Supported by: Federal Ministry for Economic Affairs and Climate Action on the basis of a decision by the German Bundestag

Methodological Approach

New HPC solver - PIPS-IPM++[6]

- We have designed custom algorithms for **distributed High Performance Computing (HPC)** to keep computing times manageable and to circumvent memory limitations
- PIPS-IPM++ implements a **parallel interior-point method** (IPM) for solving large-scale linear programs (LPs). It exploits the doubly bordered block diagonal matrix structure to parallelize the optimization process via **MPI** and **OpenMP**
  - The hourly optimization model is decomposed into time slices
- LP standard form:

$$\min\ c^T x \quad \text{s.t.}\quad Ax = b,\quad x \geq 0$$

where $c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, $A \in \mathbb{R}^{m \times n}$.

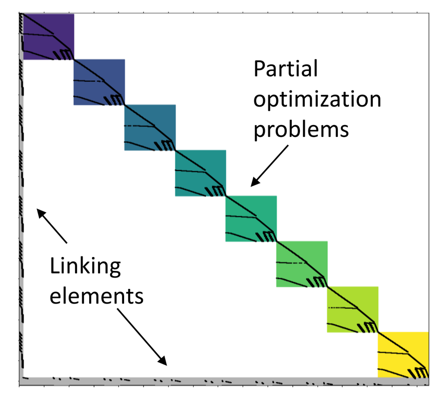

Fig. 3: Example of a typical energy system block-structured matrix

Model coupling

An optimization model has been coupled with a grid model, and an agent-based simulation to explore scenarios from different perspectives and to analyze a variety of indicators.

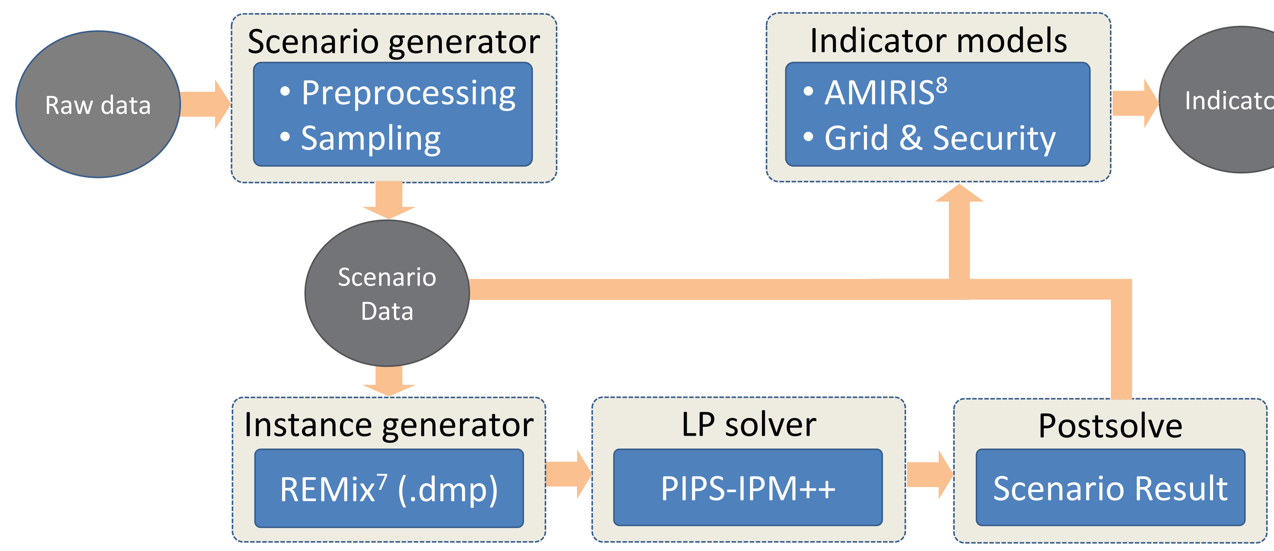

Fig. 4: High-level representation of the pipeline flow for each scenario as part of our HPC workflow

JUBE[9] – workflow environment

The automated workflow execution on HPC has been implemented based on JUBE

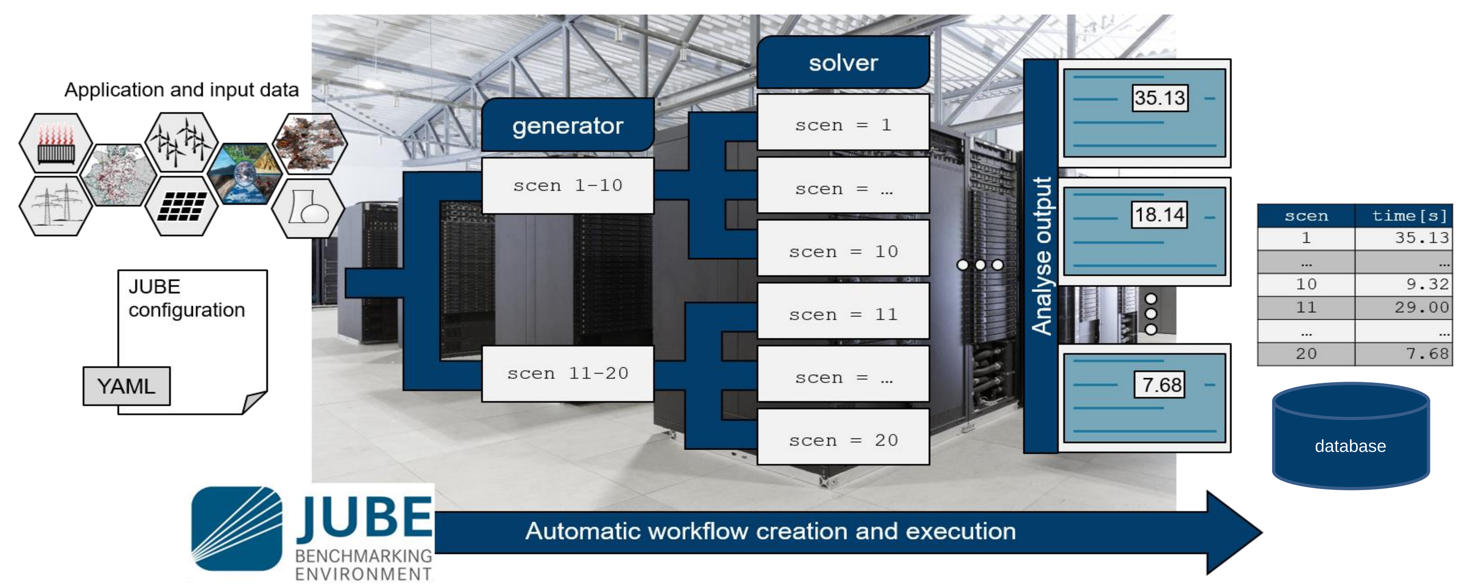

Fig. 5: Schematic representation of the HPC workflow implementation by JUBE (manages parallel pipeline flow (Fig. 4))

Technical And Scientific Results

PIPS-IPM++ scaling

Our open-source solver PIPS-IPM++[6] **outperforms state-of-the-art commercial solvers** on massively parallel architectures. A comparison on JUWELS[10]:

REMix[7] instance:
- 8.6M rows; 8.8M columns
- Up to 96 nodes; 4 threads per MPI process

PyPSA-eur[11] converted to REMix:
- 234M rows; 213M columns
- 16 nodes; 96 MPI tasks; 8 threads per task

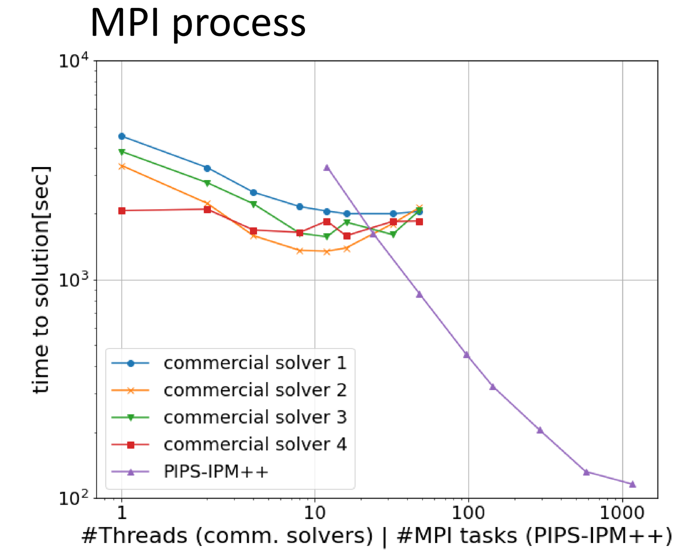

Fig. 6: REMix instance with 109 regional model nodes

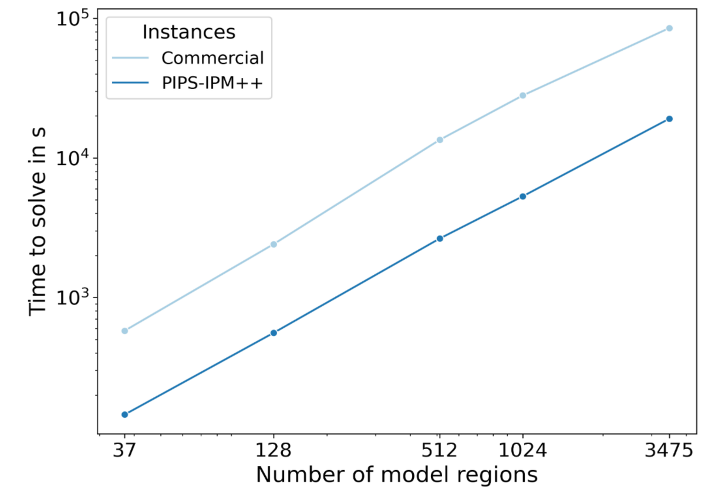

Fig. 7: REMix instance based on the PyPSA-eur dataset

Scenario analysis

- Scenario analysis based 3600 REMix instances as in Fig. 6
- Fig. 8: **Indicator evaluation:** 27 indicators build 3 clusters
  - Green (1): High share of renewables (fewer CO_2 emissions and flexibility)
  - Blue (3): High gas consumption (higher CO_2 emissions and flexibility)
  - Red (2): In between cluster 1 and 3
- Fig. 9: Photovoltaic dominates the share of power supply in most of the scenarios for 2030 with of up to 80%. The power supply by gas power plants has a wide spread between almost 0% and more than 60%. However, most of the scenarios have a rather low power supply my gas power plants.

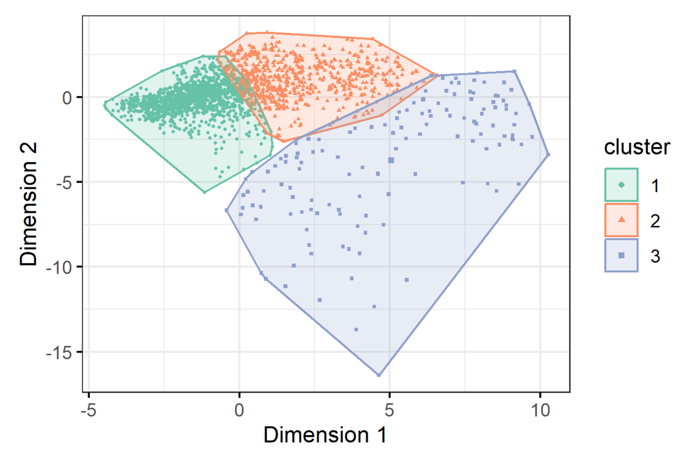

Fig. 8: Scenario cluster plot for 2 main dimensions

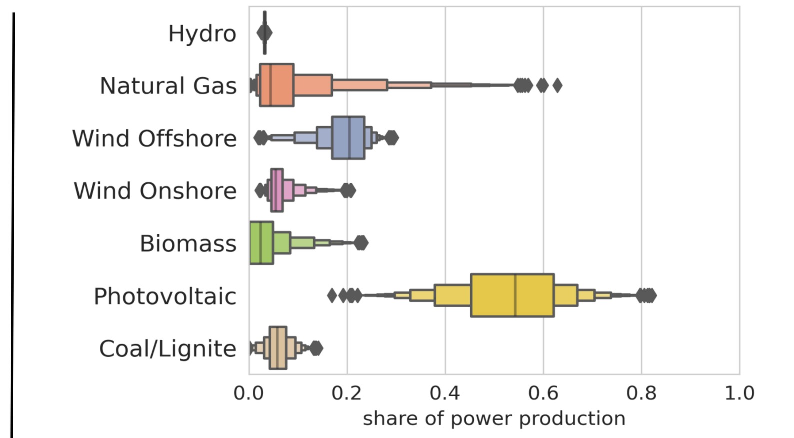

Fig. 9: Share of electricity production by source for 3600 scenarios

Outlook

Our next step is to cover more uncertainties, e.g. on weather and demand time series. These uncertainties are also considered through stochastic optimization. In this context, PIPS-IPM++ is used to solve the stochastic scenarios. Furthermore, an increase in model resolution calls for technologically more representative modeling approaches and thus, **solving mixed-integer linear programs on HPC** becomes the next challenge. We have prepared our workflow and solver software to solve such problems. We can therefore build upon our solver PIPS-IPM++ and integrate **methods for heuristics** such as those **based on a Fix-Propagate-Repair (FPR)** approach we are currently developing.

Contact: Thomas Breuer (t.breuer@fz-juelich.de)
Jülich Supercomputing Centre (JSC)

[6] https://github.com/NCKempke/PIPS-IPMpp
[7] https://www.dlr.de/ve/remix
[8] https://www.dlr.de/ve/amiris
[9] https://go.fzj.de/jsc-jube
[10] https://go.fzj.de/juwels
[11] https://github.com/PyPSA/pypsa-eur
[12] https://go.fzj.de/jsc-unseen