EGU General Assembly 2022

Inverting spatially low resolved electricity system modeling results: How feasible are they when dis-aggregated into high resolution?

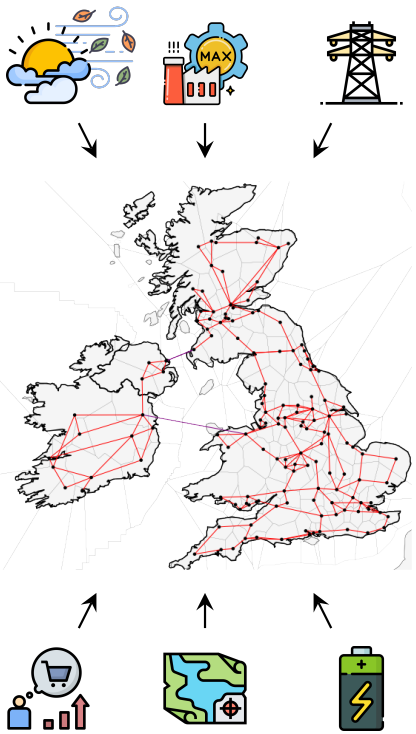
Martha M. Frysztacki[a]

May 25th, 2022

Co-authors: Prof. Dr. Tom Brown[a,b]

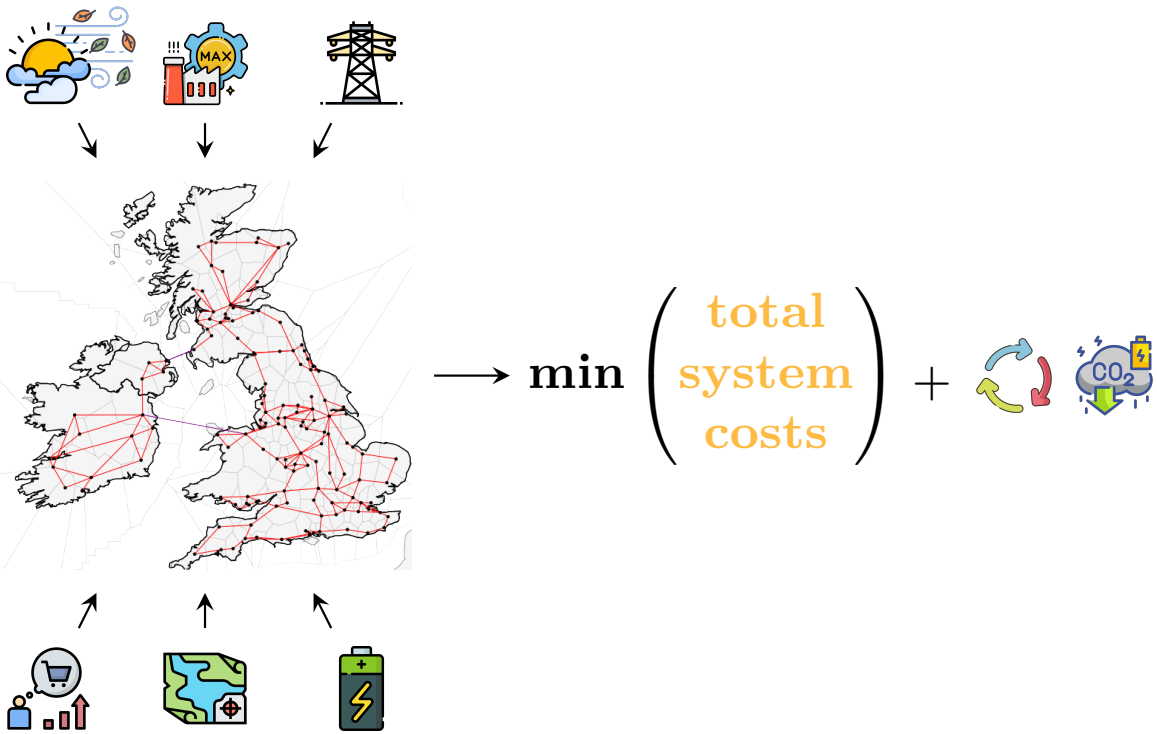
Institutions: [a]Karlsruhe Institute of Technology (KIT),
Institute for Automation and Applied Informatics
[b]Technical University of Berlin (TUB)

Study uses the open source electricity system model PyPSA-EUR.



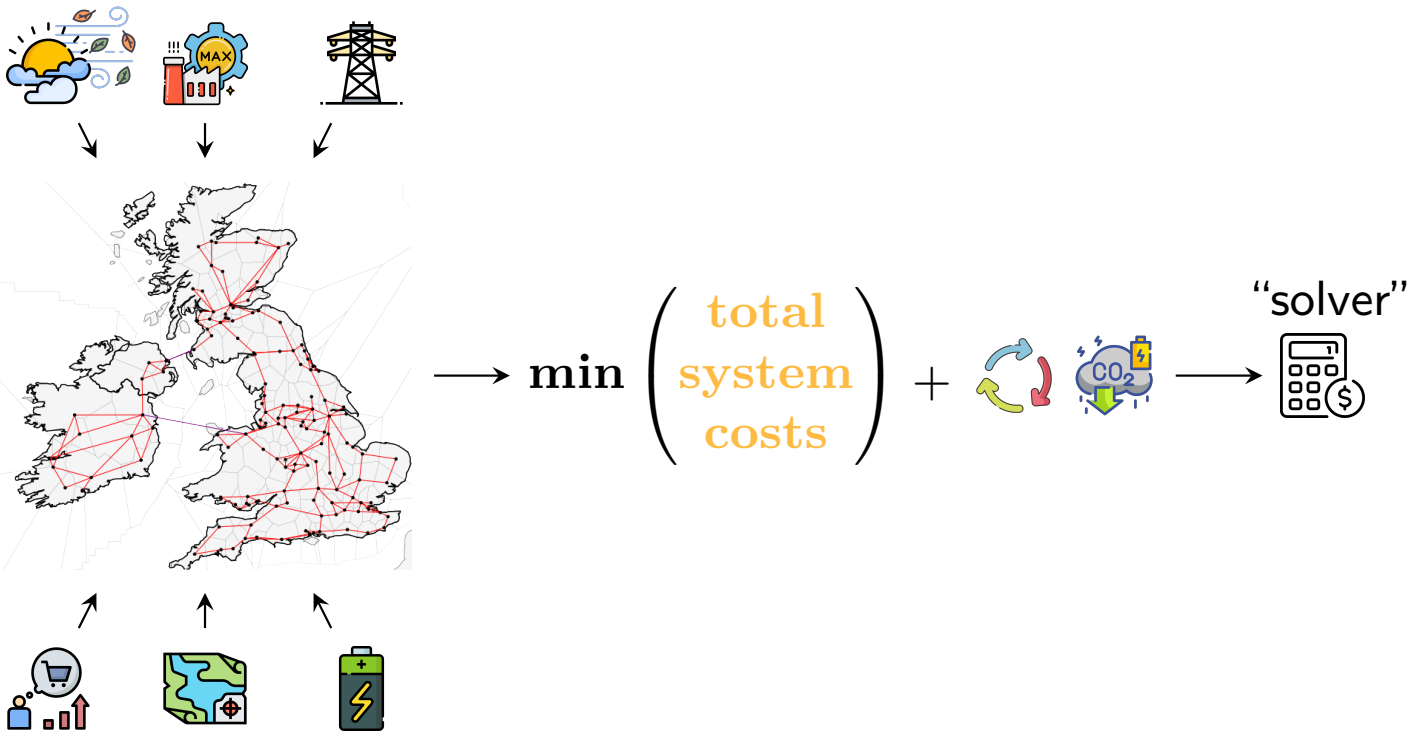
Study uses the open source electricity system model PyPSA-EUR.

PyPSA minimises the total system costs,
subject to constraints to account for climate targets,
physics & electrical laws.

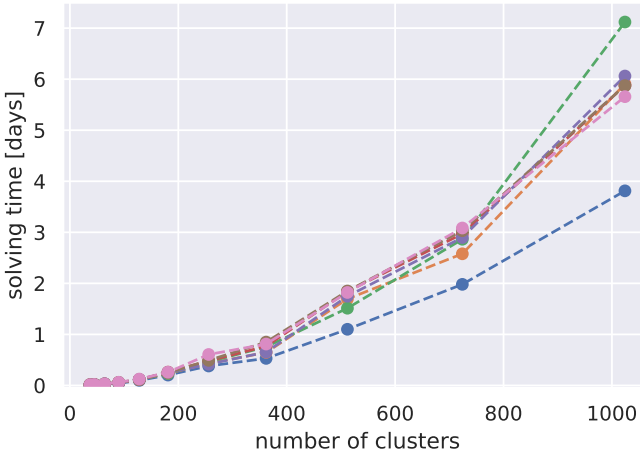


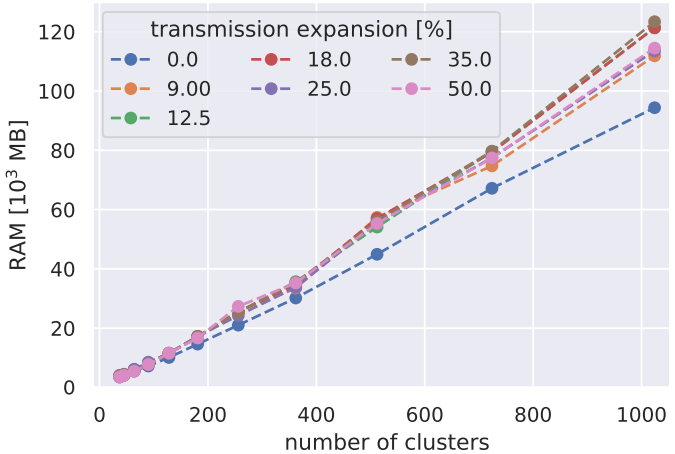
Study uses the open source electricity system model PyPSA-EUR.

PyPSA minimises the total system costs,
subject to constraints to account for climate targets,
physics & electrical laws.

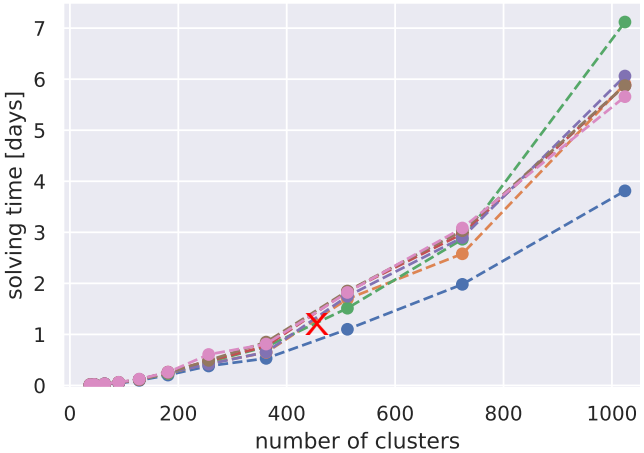


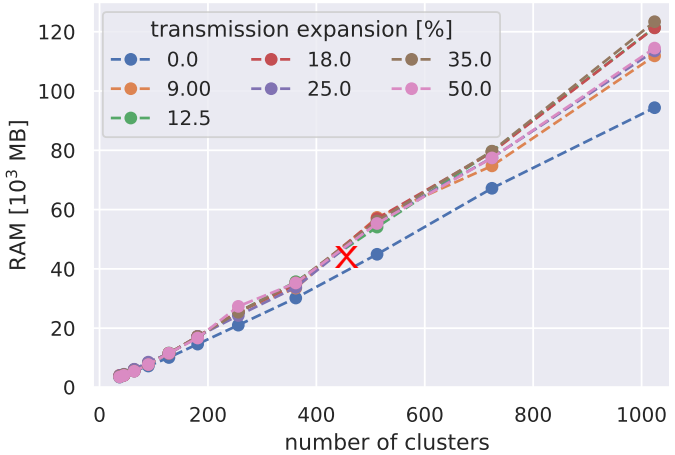
Such large optimisation problems lead to computational challenges.

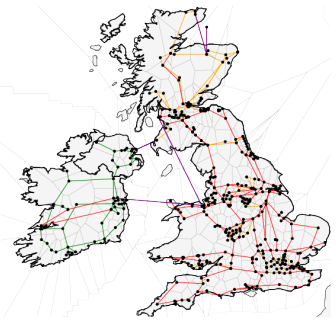




Such large optimisation problems lead to computational challenges.



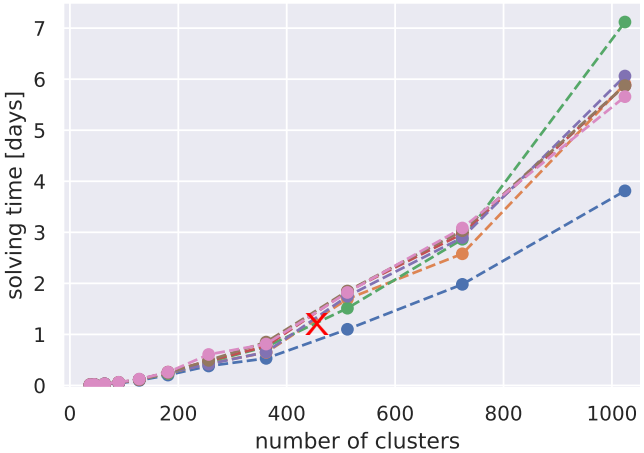


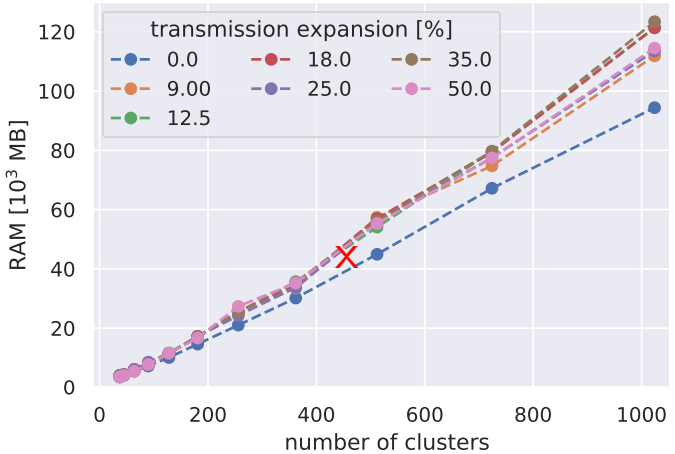


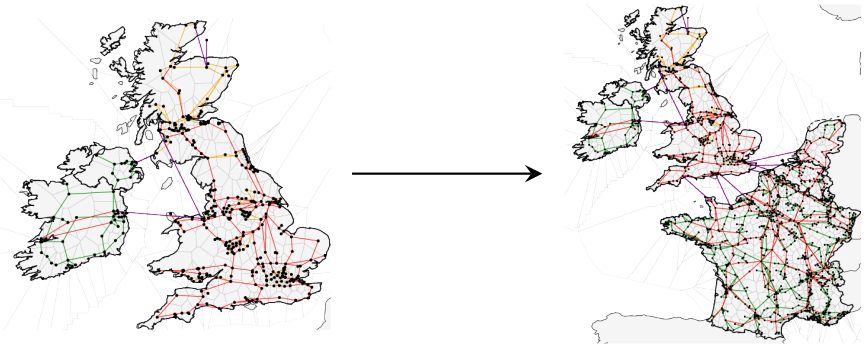
447 nodes

$\approx$ 1.5 days, 50 GB

Such large optimisation problems lead to computational challenges.







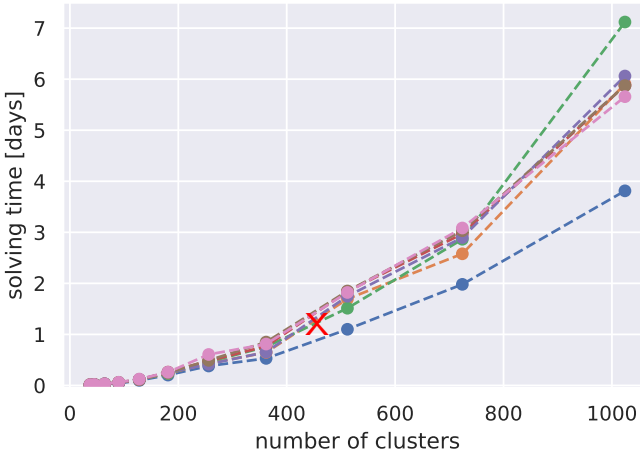
447 nodes

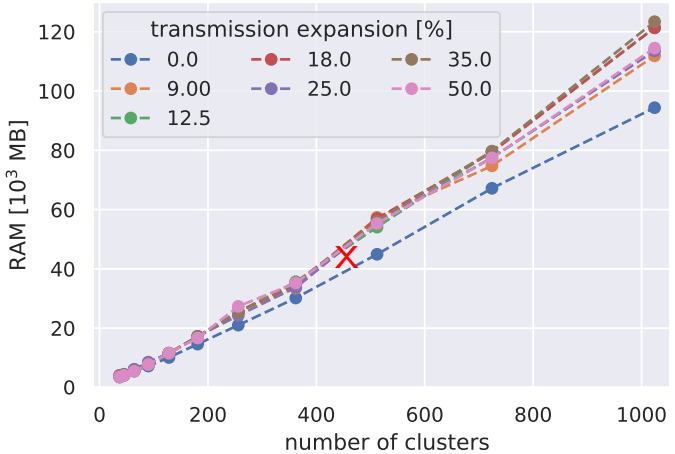
≈ 1.5 days, 50 GB

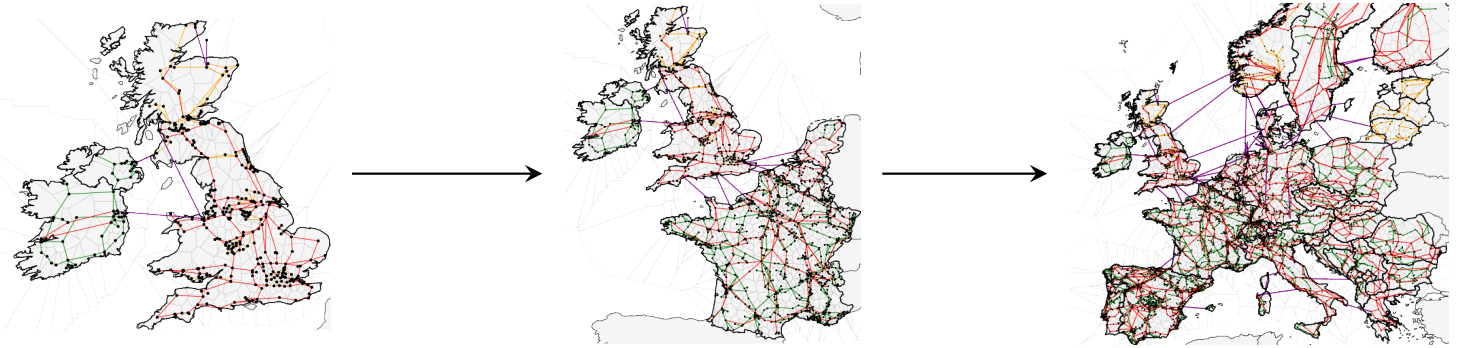
1491 nodes

≈ 10-15 days, 160 GB

Such large optimisation problems lead to computational challenges.







447 nodes

≈ 1.5 days, 50 GB

1491 nodes

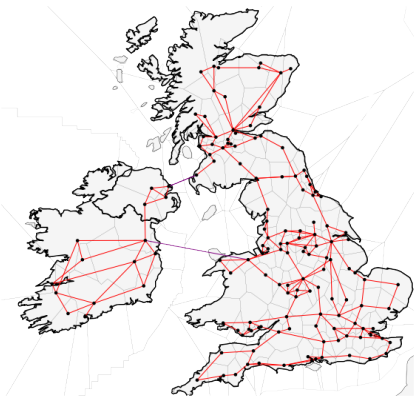
≈ 10-15 days, 160 GB

5373 nodes

≈ 50-75 days, 800 GB

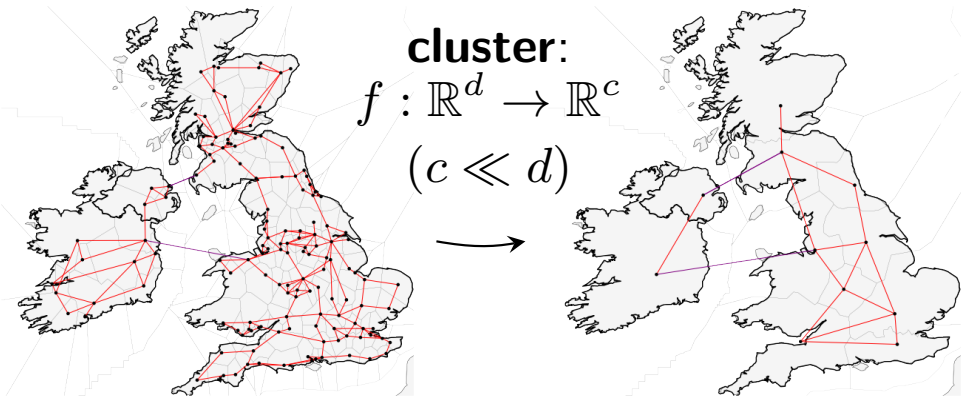
Reduce network-size $\rightarrow$ lift computational burden

Usually, the network is reduced using some kind of aggregation.



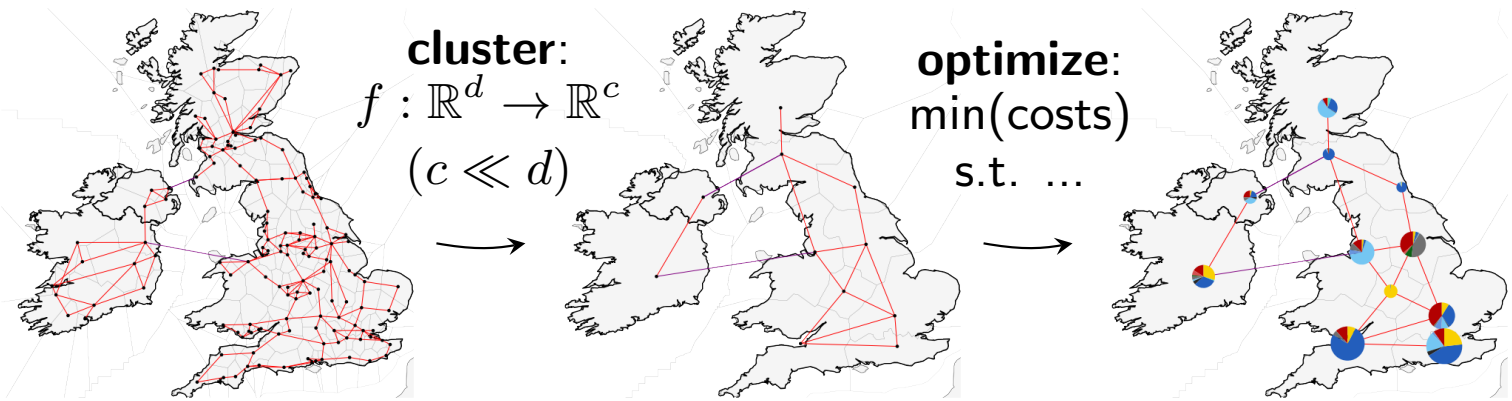
Reduce network-size $\rightarrow$ lift computational burden

Usually, the network is reduced using some kind of aggregation.



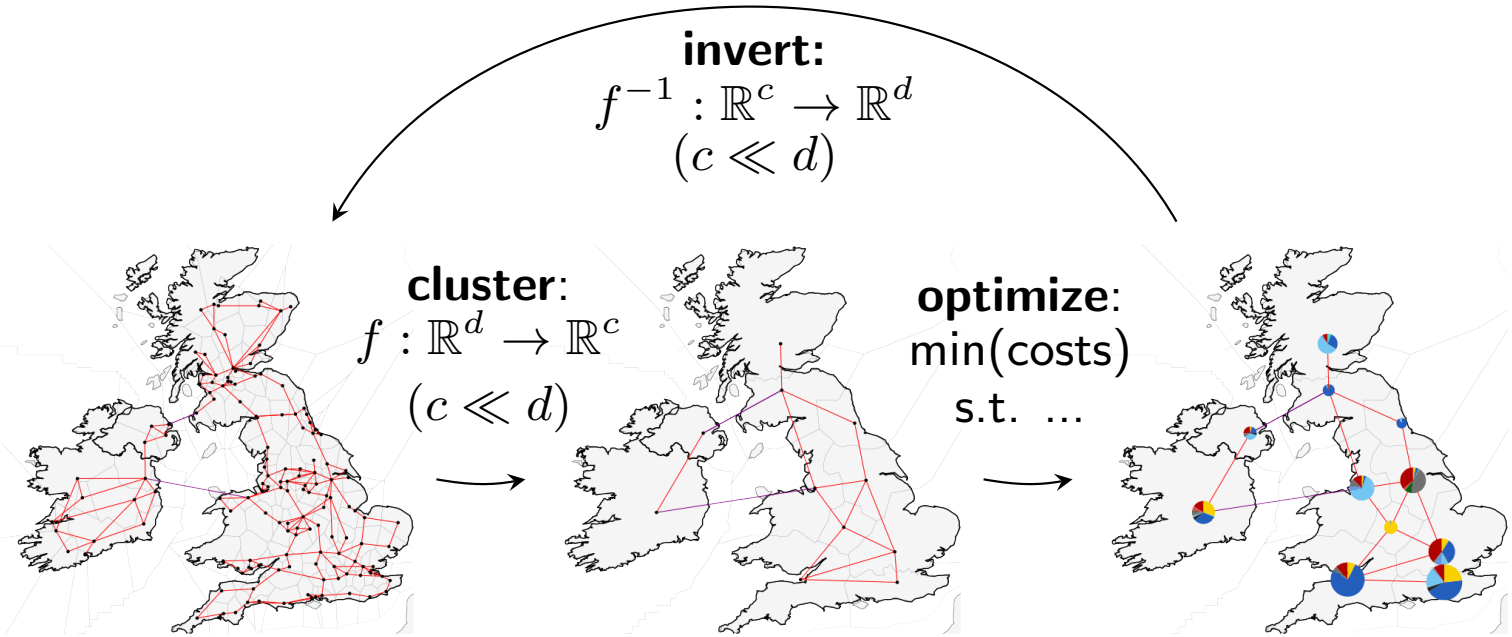
Reduce network-size $\rightarrow$ lift computational burden

And then passed to the "solver".



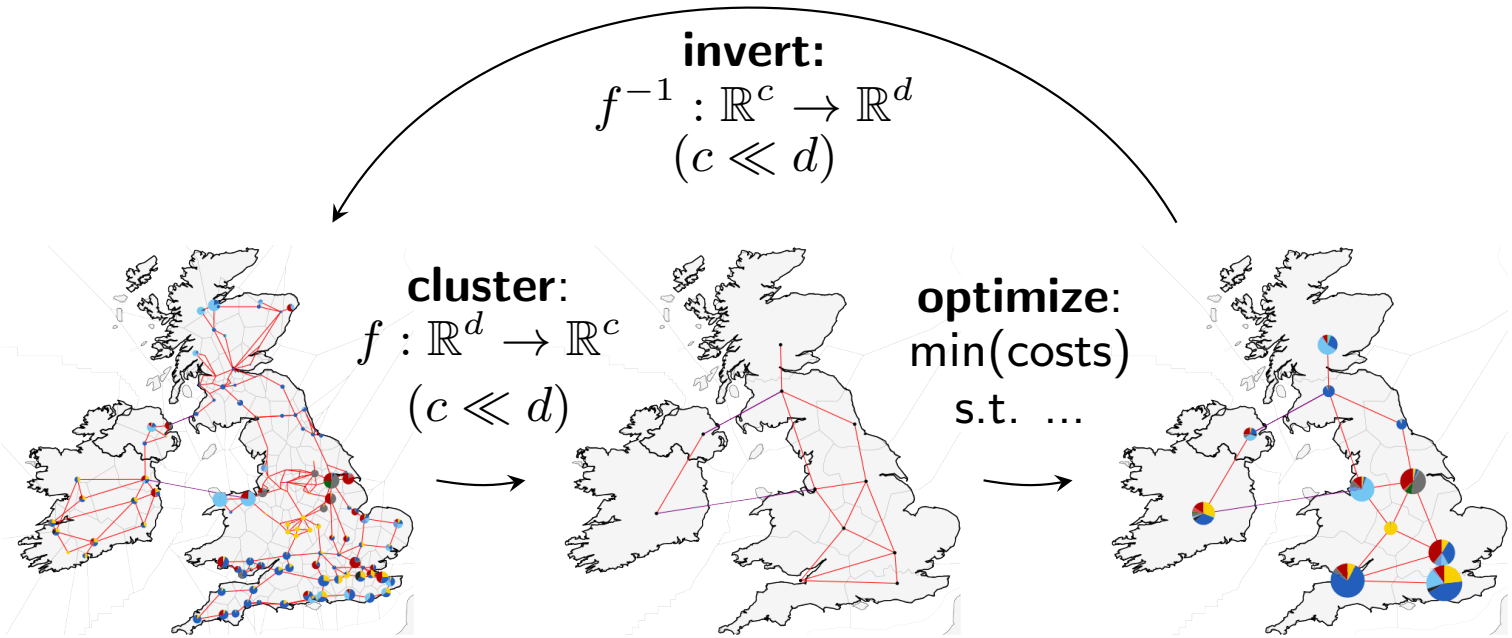
Reduce network-size $\rightarrow$ lift computational burden

But it is not clear, how feasible these results are with respect to the original problem.



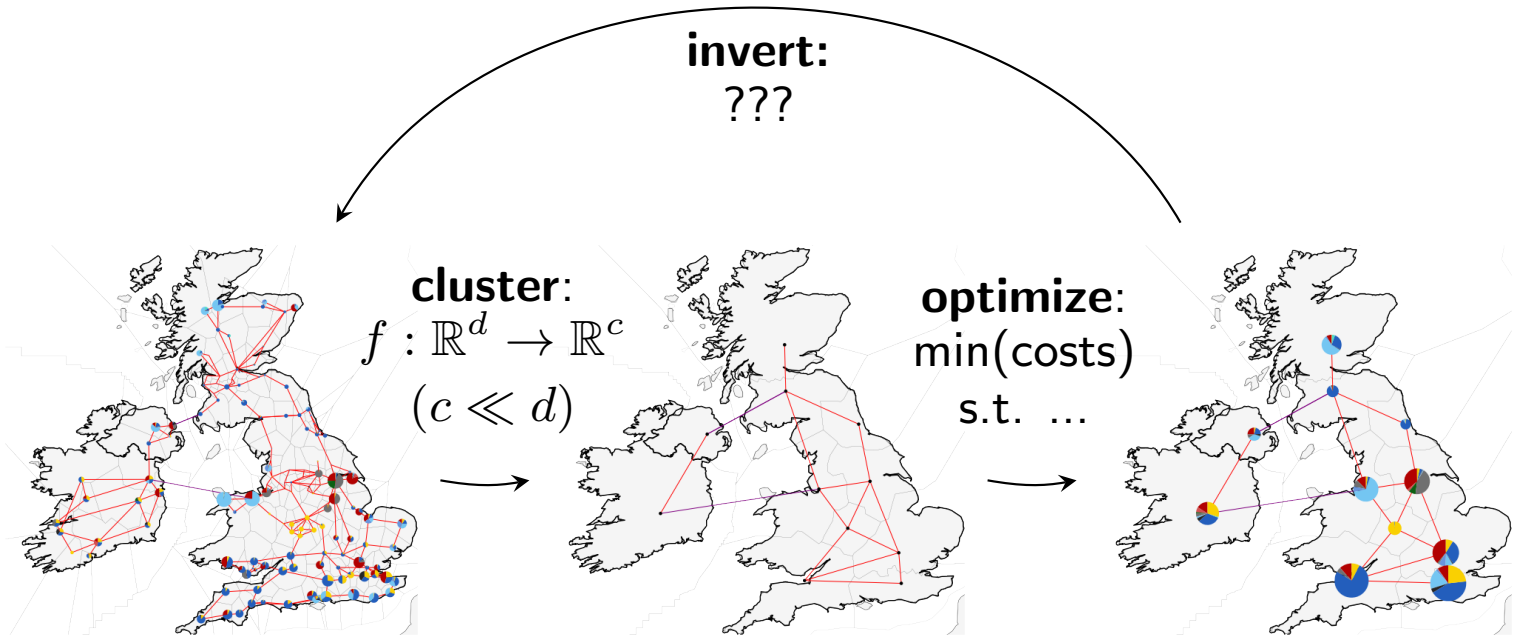
Reduce network-size $\rightarrow$ lift computational burden

And: Is there a "best way" of feeding the results back into original model?



Reduce network-size $\rightarrow$ lift computational burden

And: Is there a "best way" of feeding the results back into original model?



Three approaches to dis-aggregate results: 1. uniform

Abbreviation	Explanation
uniform	**uniformly distribute** resulting optimal capacity of a cluster across associated high-resolution nodes
re-optimized	
min excess	

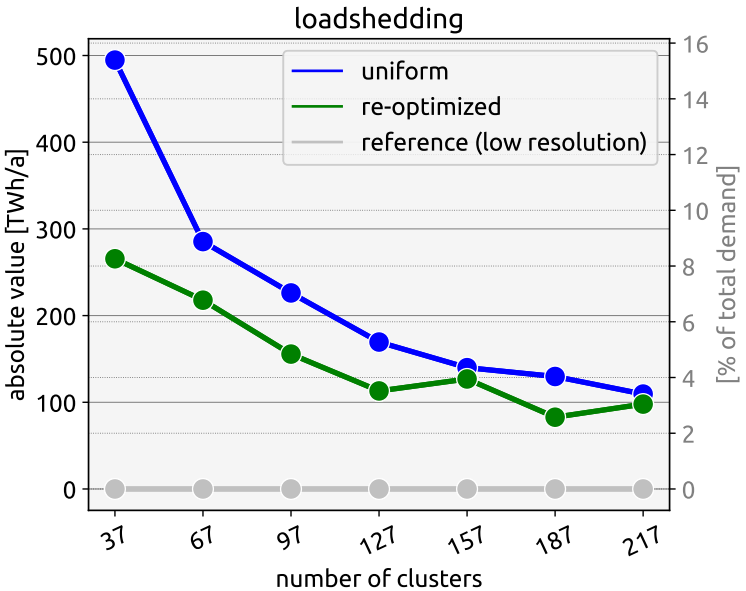
Three approaches to dis-aggregate results: 2. re-optimized

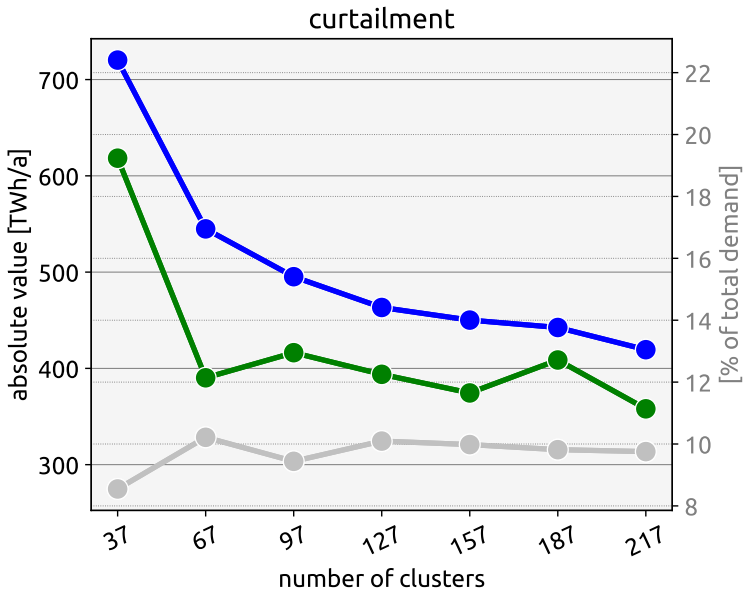
Abbreviation	Explanation
uniform	**uniformly distribute** resulting optimal capacity of a cluster across associated high-resolution nodes
re-optimized	within each cluster, solve the **full minimization problem** with additional constraint that total capacities of reference model and high resolution model agree.
min excess	

Three approaches to dis-aggregate results: 3. min excess

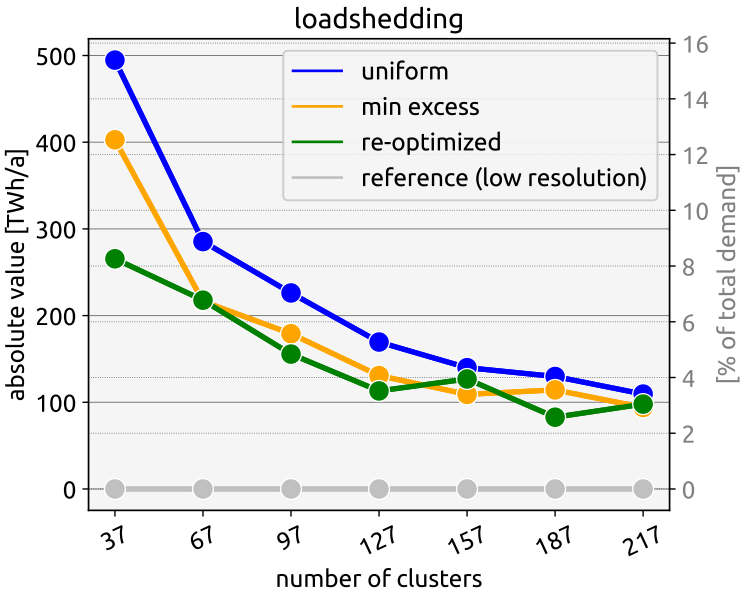
Abbreviation	Explanation
uniform	**uniformly distribute** resulting optimal capacity of a cluster across associated high-resolution nodes
re-optimized	within each cluster, solve the **full minimization problem** with additional constraint that total capacities of reference model and high resolution model agree.
min excess	minimize "excess electricity" (**custom objective**) within each cluster such that total capacities of reference model and high resolution model agree.

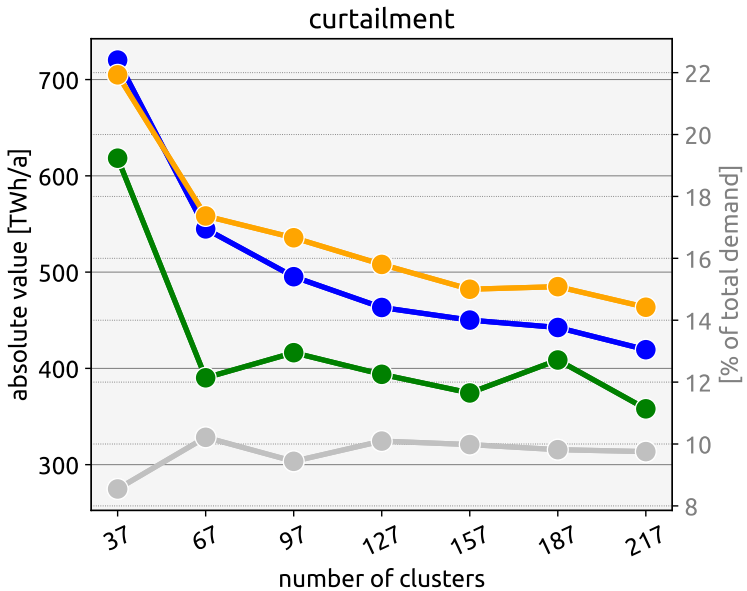
Trade-Offs of the three approaches to dis-aggregate results.



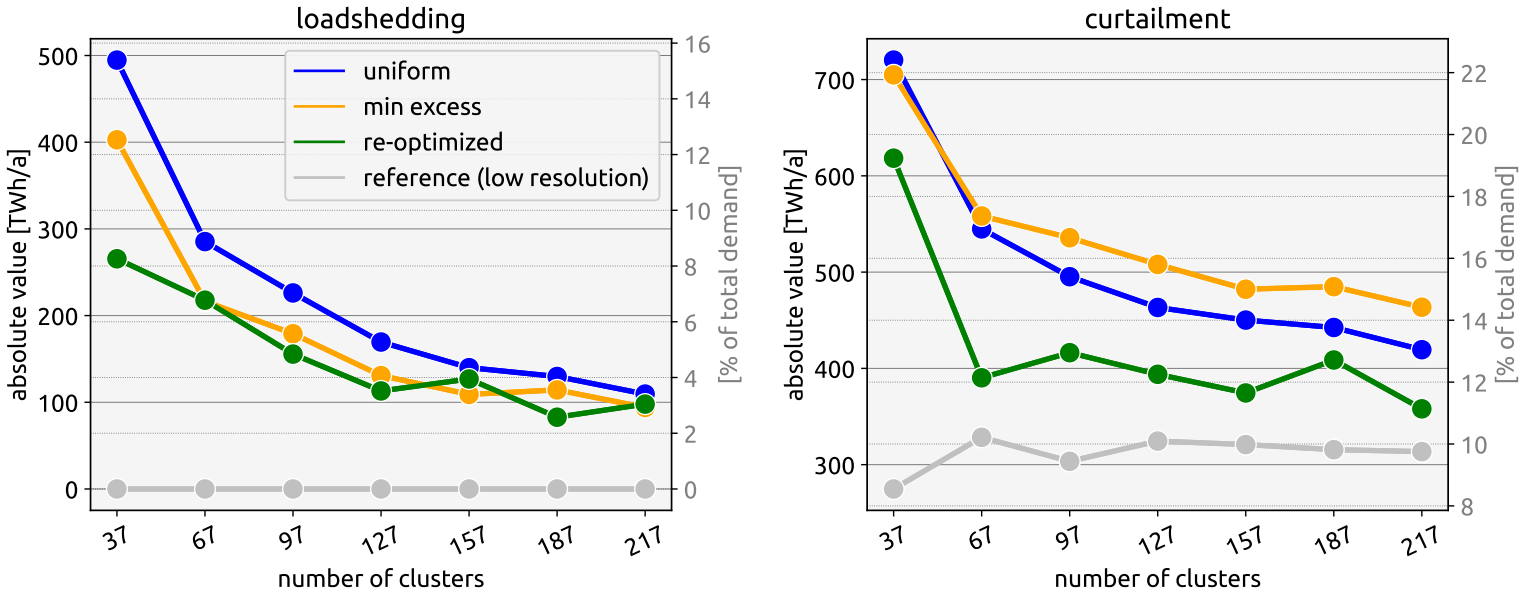


Trade-Offs of the three approaches to dis-aggregate results.



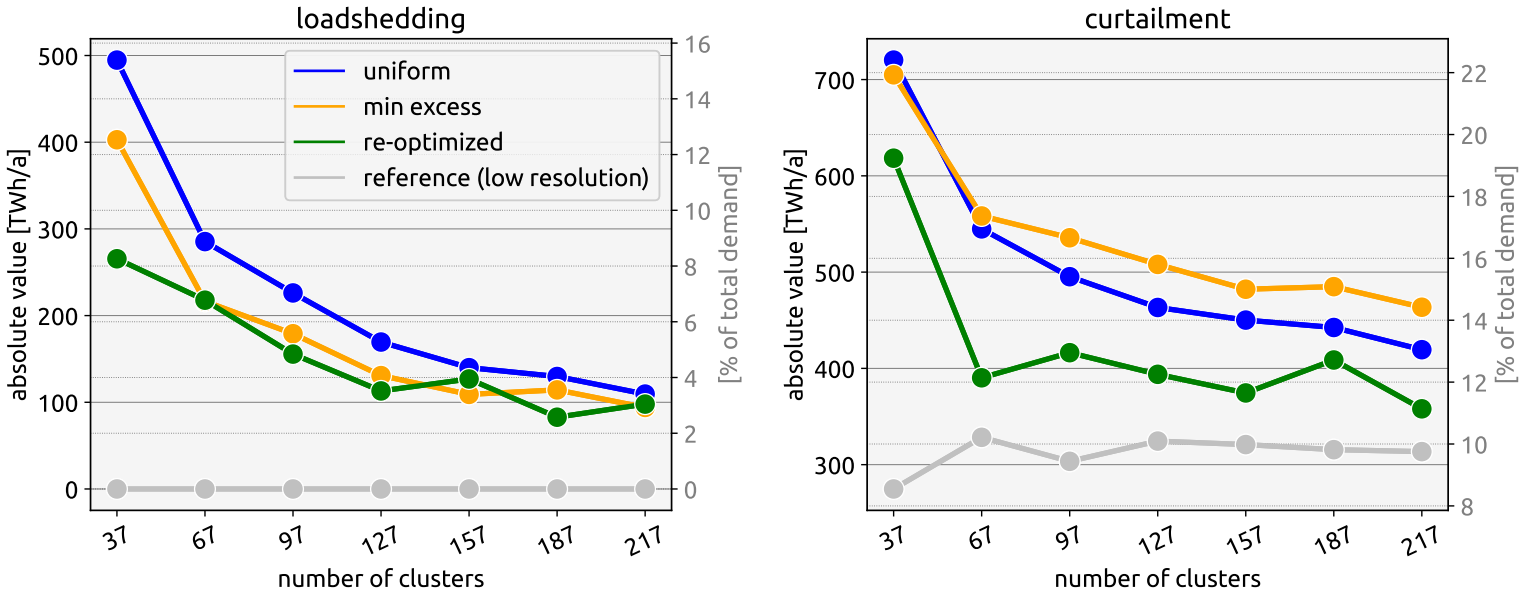


Trade-Offs of the three approaches to dis-aggregate results.



$\Rightarrow$ Load shedding in "min excess" similar to "re-optimized" results and $1-5\%$ lower than in "uniform". $(++)$

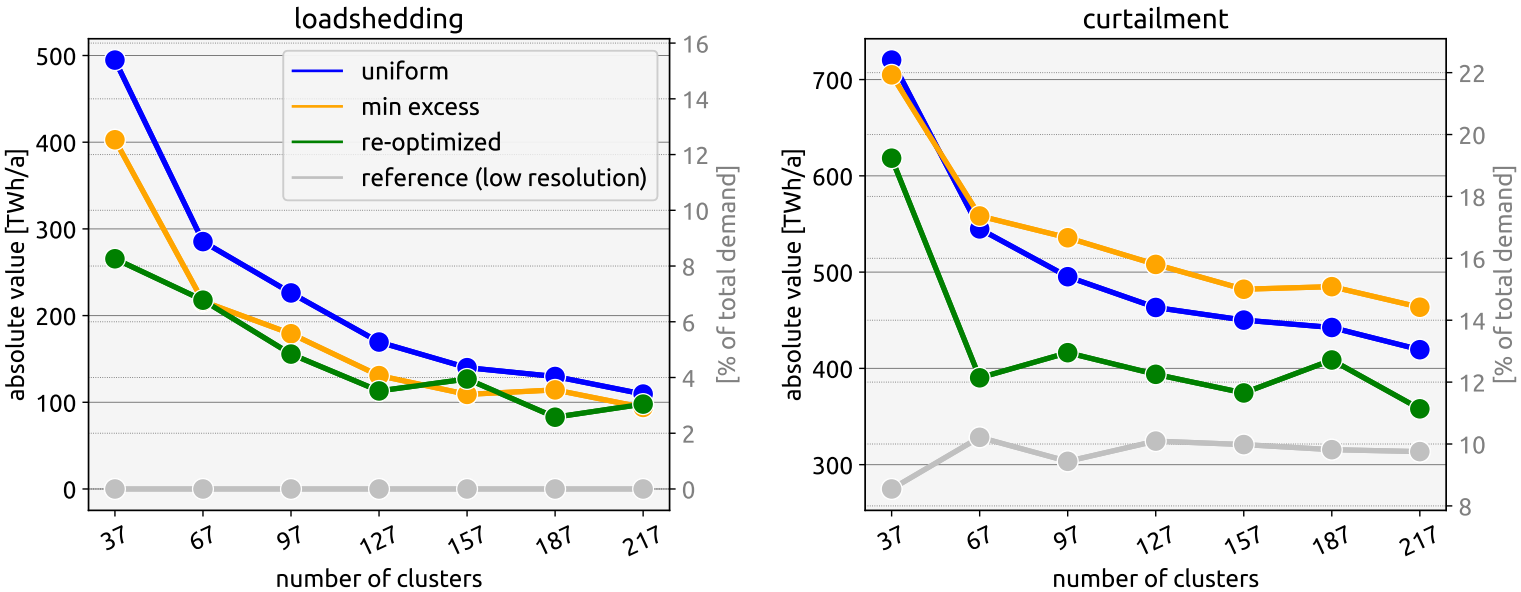
Trade-Offs of the three approaches to dis-aggregate results.



⇒ Load shedding in "min excess" similar to "re-optimized" results and $1-5\%$ lower than in "uniform". $(++)$

⇒ Curtailment in "min excess" $1-2\%$ higher than in "*uniform*" and $4-7\%$ higher than in "re-optimized". (-)

Trade-Offs of the three approaches to dis-aggregate results.

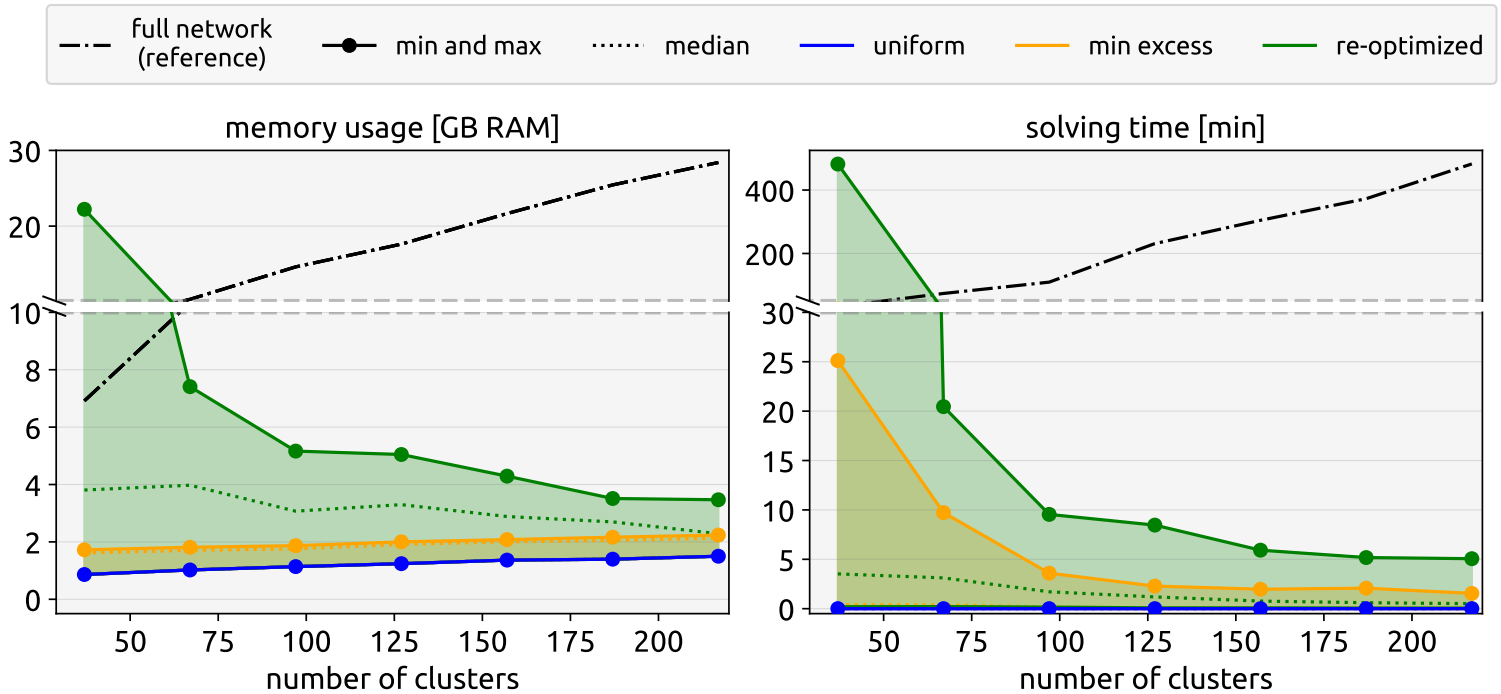


⇒ Load shedding in "min excess" similar to "re-optimized" results and $1-5\%$ lower than in "uniform". (++)

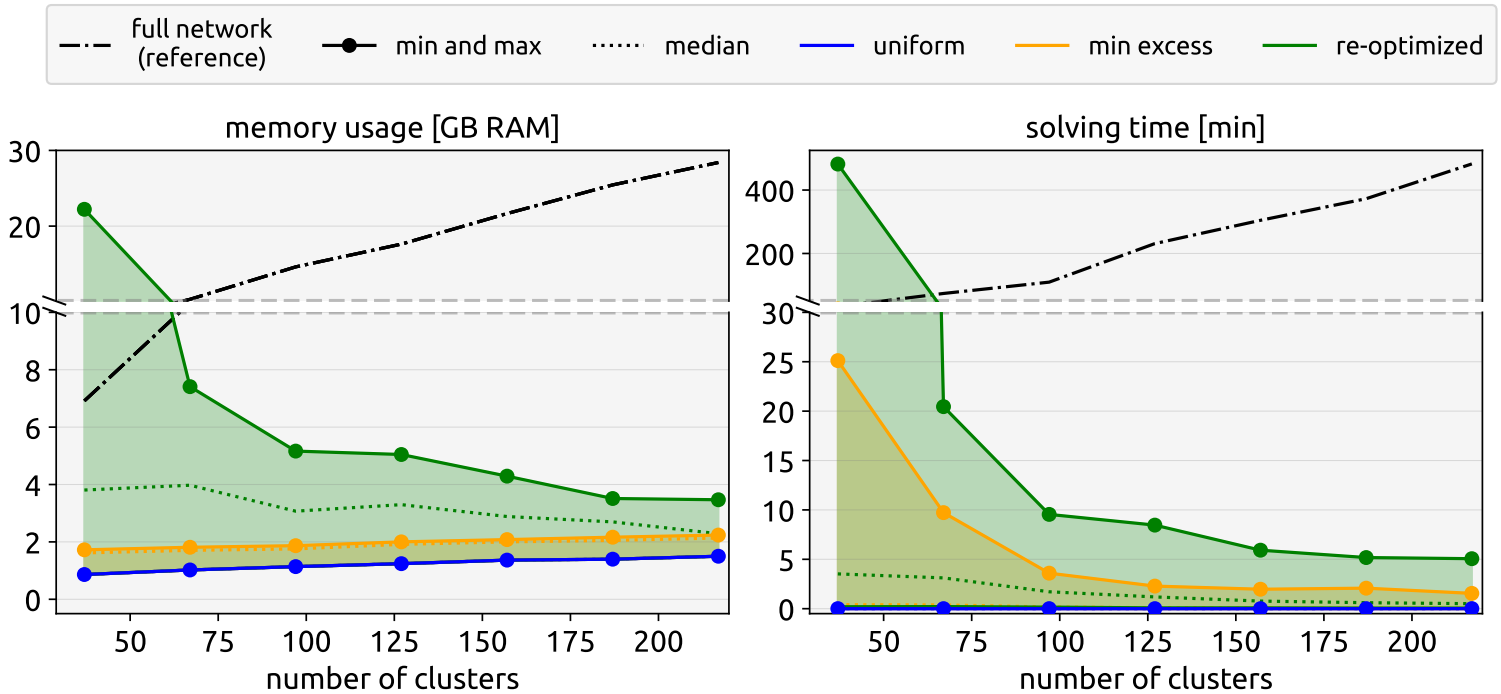
⇒ Curtailment in "min excess" $1-2\%$ higher than in "*uniform*" and $4-7\%$ higher than in "re-optimized". (-)

⇒ **Dis-aggregated results of a 200 nodes reference model can not cover 2-4% of electricity demand, regardless of the dis-aggregation method.**

Trade-Offs of the three approaches to dis-aggregate results.



Trade-Offs of the three approaches to dis-aggregate results.



⇒ Re-optimizing can result in heavy computational burden, in few cases heavier than the original problem.

Trade-Offs of the three approaches to dis-aggregate results.

	Implementation	Solving Time	Memory (RAM)	Results Quality
uniform	✓	✓	✓	✗
min excess	✗	✓	✓	✓
re-optimize	✗	✗	✗	✓

✓ indicate a reasonable trade-off, entries with a ✗ indicate an inadequate compromise.

Thank you for your attention... Questions?

License

Back-Up Slides

Outlook (Future Research).

- Research possibly better suited inverse functions.
- Adapt all proposed dis-aggregation methods to include transmission expansion.
- For *min excess*: more sophisticated distribution of storage, e.g. include in objective function.
  Currently, $G_{n,\mathrm{H2}} \sim \mathrm{G}_{n,\mathrm{onwind}}$ and $G_{n,\mathrm{battery}} \sim \mathrm{G}_{n,\mathrm{solar}}$. Better distribution of storage could avoid more load shedding and curtailment.
- Adapt all these dis-aggregation methods to include carbon budgets or develop new ones. We only considered a carbon budget of 0.

Three approaches to dis-aggregate results.

Abbreviation	Explanation
uniform	uniformly distribute the obtained capacity of a cluster c to its associated nodes, i.e. $$G_{c,s} \mapsto \frac{1}{\lvert N_c \rvert} \begin{pmatrix} G_{c,s} \\ \dots \\ G_{c,s} \end{pmatrix} \in \mathbb{R}^{\lvert N_c \rvert}$$
min excess	minimize excess energy within each cluster c, i.e. $$\min_{G_{n,s}} \sum_{\substack{n \in N_c, \\ s,t}} \left[\bar{g}_{n,s,t} G_{n,s} - d_{n,t} - 0.7 \sum_{\substack{l_{(i,j)} \in \mathcal{L}: \\ i=n \vee j=n}} F_{(i,j)} \right]^+$$ $s.t.\ \sum_{n \in N_c} G_{n,s} = G_{c,s}$ and $G_{n,s} \leq G_{n,s}^{\max} \quad \forall n \in N_c$.
re-optimized	within each cluster c, solve the full minimization problem with the additional constraint $\sum_{n \in N_c} G_{n,s} = G_{c,s}$.

Data I

usage	source	details
land use	CORINE	https://land.copernicus.eu/pan-european/corine-land-cover/clc-2012?tab=metadata
land use	Natura 2000	https://www.eea.europa.eu/data-and-maps/data/natura-10#tab-metadataa
offshore regions	EEZ	http://www.marineregions.org/disclaimer.php
country & synchronous zone borders	NUTS3	https://ec.europa.eu/eurostat/web/gisco/geodata/reference-data/administrative-units-statistical-units
time series demand	OPSD	https://data.open-power-system-data.org/time_series/2019-06-05/README.md
time series wind	atlite (ERA5)	https://github.com/PyPSA/atlite
time series solar	atlite (SARAH-2)	https://github.com/PyPSA/atlite