

Table of Contents

1	<i>Objective of the Workshop</i>	2
2	<i>InnOpTEM Project Overview and Modeling Approaches</i>	4
2.1	Introduction Julian (Amprion) for InnOpTEM	4
2.2	Project consortium:	4
2.3	Technical Setup	4
2.4	Modeling Approaches Compared	4
2.5	Resulting Outputs	5
2.6	Julian – On Data Challenges	5
2.7	Example:	5
2.8	Constraint Challenges	5
3	<i>Experience from PSE – A Journey Through Practical Switching Tools</i>	5
3.1	Endika (PSE – Polish TSO)	5
3.2	Approach 1: MILP with Gurobi	6
3.3	Approach 2: Iterative Ranking via Linear OTS	6
3.4	Approach 3: Generalized LODF (Line Outage Distribution Factor)	6
3.5	Handling N–1 Security	6
3.6	Daily Decision Support: From Hourly to Day-Ahead Plans	6
4	<i>50Hertz/Elia: High-Speed GPU-Based Topology Optimization</i>	7
4.1	Presented by Nico (50Hertz)	7
4.2	Objective: Speed and Scalability	7
4.3	Tool Features and Data Sources	7
4.4	Key Innovation: GPU-Based Genetic Algorithm	7
4.5	What We Optimize	7
4.6	Constraints & Realism	8
4.7	Validation and Filtering	8
4.8	Summary of Benefits	8
5	<i>Interactive Discussion – Constraints, Trust, and Deployment Realities</i>	8
5.1	Facilitators: Julian (Amprion), Christian (RWTH), and others	8
5.2	Topic 1: Real-World Constraints for Switching Actions	8
5.3	Topic 2: Modeling Limitations and AI Use	9
5.4	Topic 3: Practical Considerations and Parallelization	9
5.5	Topic 4: Benchmarking and Data Access	9
6	<i>Live Demo – Julia, PowerModels.jl, and Custom Extensions</i>	9
6.1	Setup and Tools Used	9
6.2	Demo Steps	10
6.3	Optimization Techniques	10
6.4	💡 Performance Highlights	10
6.5	Closing Notes from RWTH	10
7	<i>Session Wrap-Up and Reflections</i>	10
7.1	Key Observations	10
7.2	Research and Industry Gaps	11
7.3	Final Recommendations	11
8	<i>Closing Julian (Amprion)</i>	11

Here is a structured summary of the Topology Optimization Workshop (June 25, 2025) held at the Energy Innovation Summit in Berlin, including key learnings, strategic takeaways, and actionable follow-ups:

1 Objective of the Workshop

To share practical experiences, challenges, and solution strategies for topology optimization (aka optimal transmission switching) from European TSOs and research institutions, and to align on realistic pathways for implementation and evaluation.

Participating Organizations

- TSOs: Amprion (DE), PSE (PL), 50Hertz (DE), Elia (BE)
- Research: RWTH Aachen (IAEW and OR), FGH
- Audience: Other TSOs, RCCs, researchers, tool developers

Key Takeaways

a) Speed, Trust, and Usefulness Must Align

- Too aggressive = operator rejection.
Early models optimized perfectly but lacked acceptance due to unrealistic switching volumes or risk profiles.
- Support tools, not black boxes: Tools must provide alternatives, not single solutions.
- Operator trust depends on constraint completeness, interpretability, and fallback options.

b) Modeling Approaches Are Complementary, Not Competitive

Approach	Strength	Weakness
MILP + Gurobi	Accurate, comprehensive	Slow, inflexible
Sensitivity-based iteration	Fast, tunable	Approximate, partial
Generalized LODF	Very fast, scalable	Limited switching complexity
GPU-based Genetic Algorithm	Ultra-fast, tunable	Risk of over-aggression
AI/ML (experimental)	Learns patterns	Limited generalizability

Conclusion: A hybrid architecture is best — fast preselection + robust validation.

c) Security Constraints: Smarter Filtering Wins

- Full N–1 evaluation is infeasible in large grids.
- Practical heuristic: Only check contingencies with >90% base-case loading.
 - In Poland: reduced from 400 to ~21 relevant contingencies
 - In Germany: similar filtering helped keep runtime <20 min

d) Substation Modeling and Realism

- UCTE/CGMES models are bus-branch, lacking:
 - Switch types
 - Node-breaker detail
 - Static element IDs
- These gaps lead to unrealistic solutions unless pre-processed.

→ Each TSO built custom rules to “guess” real topology — prone to error and not scalable.

e) AI's Role is Assistive — Not Autonomous (Yet)

- AI can:
 - Filter likely branches
 - Prioritize actions based on historical context
- AI cannot:
 - Guarantee feasibility or robustness
 - Handle rare but critical system states

→ Position AI as aids to human-guided or hybrid solvers, not replacements.

f) Urgent Need for Open Benchmarks

- No shared dataset with:
 - Full substation topology
 - Load/generation realism
 - Contingency definitions
- All participants agreed: this blocks research, slows development, and prevents method comparison.

→ Opportunity for a community-driven benchmark initiative, possibly via ENTSO-E, RTE, or academia.

g) Operational Learnings

Constraint	Recommendation
Max switch actions	Keep ≤ 3 per hour
Switch types	Favor circuit breakers; avoid disconnectors in unmanned substations
Timing	Allow day-ahead proposals, but execution timing remains a real-time decision
Aggressiveness	Tune during post-processing (e.g., trade-off curve UI)
Validation	Always post-filter with AC load flow, even if DC used in core optimization

h) Actions & Recommendations

For TSOs:

- Start using hybrid tools that combine MILP/heuristics + post-filtering
- Share anonymized data (where possible) with research partners
- Work together (e.g., under ENTSO-E umbrella) to define a public benchmark model

For Researchers:

- Focus on model trustworthiness, not just optimality
- Build modular, GPU-compatible optimization workflows
- Wrap tools into UI-based proposals — not notebooks

For Developers:

- Make switching proposals tunable (e.g., 3–5 alternatives, varying aggressiveness)
- Use dual flows or overload energy as proxy redispatch cost metrics
- Add AC-based constraint screening as post-solve filter

What's Next?

A follow-up symposium on topology optimization will be hosted in September 2025 by the German TSOs.

2 InnOpTEM Project Overview and Modeling Approaches

2.1 Introduction Julian (Amprion) for InnOpTEM

Our goal was to build and test realistic topology optimization approaches using actual data and operator constraints. That's why we focused on the day-ahead timeframe — because that's where such a tool could support real-world operation. We also wanted to identify which constraints are most critical to include so that operators can trust and use the results.

We aimed for an approach that runs within one to three hours, which is necessary for practical day-ahead planning.

2.2 Project consortium:

- Amprion (Germany) – TSO and system operations lead
- 4 TSOs Germany – participating TSO
- FGH and RWTH Aachen – research partners
 - IAEW institute (specialized in power systems and operations research)
 - OR institute

We combined knowledge in system operations, optimization, and simulation to design and benchmark different modeling strategies.

2.3 Technical Setup

We had to build a custom data processing pipeline because the typical CGM (Common Grid Model) or UCTE congestion forecast models don't include the full range of information needed for topology optimization.

We gathered:

- Grid data
- Power plant data
- PST (phase-shifting transformer) behavior
- Technological degrees of freedom

From this, we constructed a database enabling four distinct optimization approaches:

2.4 Modeling Approaches Compared

i) Integrated MILP Formulation (RWTH Aachen + Google OR-Tools)

- Full mixed-integer linear formulation
- Very accurate — considers nearly all constraints
- Uses benefit positioning and heuristics
- Limitation: Computation time — not always feasible within the 1–3 hour window

j) Iterative Sensitivity-Based Optimization (FGH)

- Based on linearizing effects of switching actions
- Faster and AC-load flow compatible
- Uses dual variables to iteratively refine switching choices
- Limitation: Cannot model all nonlinearities or very complex interactions

k) AI-Based Multi-Agent Deep Reinforcement Learning (RWTH)

- Experimental approach to “learn” switching heuristics
- Promising long-term potential
- Limitation: Currently too immature for national-scale use

l) Core Optimization-Based Topology Preselection

- Precomputes promising topologies and uses them in combined optimization
- Fast and scalable
- Limitation: Less flexibility for unanticipated network situations

2.5 Resulting Outputs

All approaches provided:

- Final unit commitment
- Integrated dispatch + switching actions
- Optional PST adjustments

We aimed to ensure all key operational constraints were respected.

2.6 Julian – On Data Challenges

One of the biggest blockers was data quality and format.

We primarily worked with UCTE congestion forecast models, which use a bus-branch representation. These models:

- Lack metadata on switching options
- Contain inconsistent naming (start/end node IDs vary over time)
- Provide implausible current limits and no short-circuit data

We had to build an extensive pre-processing step to recover a realistic and stable network model suitable for optimization.

2.7 Example:

One substation in the UCTE model listed 11 “220 kV” busbars — but in reality, it has just three 380 kV busbars. Without correction, the model allows excessive freedom and leads to unrealistic solutions.

2.8 Constraint Challenges

Some constraints are difficult to model precisely:

- Hydro plant coupling constraints
- Voltage magnitude constraints
- Short-circuit constraints (absent in UCTE data)
- Switching limits based on safety or operator policies

In practice, there's always a trade-off between:

- Solution time
- Accuracy
- Robustness

Not all approaches excel in all three. The goal is to strike a useful balance that helps system operators make informed, feasible decisions.

3 Experience from PSE – A Journey Through Practical Switching Tools

3.1 Endika (PSE – Polish TSO)

Hello everyone, I'm Endika from PSE. I'll share some of our insights and challenges from the past three years developing topology optimization tools.

We haven't reached the end of our journey yet — we're still refining the tool — but I'd like to present our thinking so far, especially on:

- Different approaches we tested
- How we handle the N-1 criterion
- And how we structure the daily topology decisions

We always saw this as a support tool, not a fully automated replacement. That's why it doesn't need to be 100% accurate — but it must be fast, robust, and give the operator meaningful suggestions.

3.2 Approach 1: MILP with Gurobi

Our first idea was straightforward: use Mixed Integer Linear Programming (MILP) and solve it with Gurobi.

Pros:

- Precise modeling of topological decisions
- Clean, optimal solutions

But: MILP gives only one solution, which can be too rigid for operators. Sometimes the optimal answer isn't what they'd actually implement — so it's better to offer a list of 10 viable actions to choose from.

We also ran into computation time barriers, especially when adding the N-1 security constraints.

3.3 Approach 2: Iterative Ranking via Linear OTS

Our second strategy was to:

- Pre-select a subset of switching actions
- Evaluate their power flow impact
- Rank them iteratively using dual variables from the linear OTS

This method let us overestimate the congestion impact and select the most promising actions quickly. It's faster than MILP and avoids needing to evaluate all flows fully.

Still, when scaling to the ENTSO-E grid, computation time again became an issue.

3.4 Approach 3: Generalized LODF (Line Outage Distribution Factor)

We developed a generalized version of the LODF matrix, which allowed us to:

- Evaluate how switching combinations affect power flows
- Avoid computing full load flows during runtime
- Precompute the matrix ahead of time (e.g., 4 hours in advance)

This dramatically reduced runtime: final switching proposals could be computed in under 20 minutes, enabling operator use in a 45-minute decision window. But there's a key assumption: we limit topological changes to max. 3 switching actions. That's what operators are willing to consider.

3.5 Handling N-1 Security

Full evaluation of 400 contingencies per switching combination was infeasible.

So, we used a practical heuristic: only consider contingencies where at least one branch is close to overload (e.g., above 90% utilization). That gave us about 21 critical contingencies on worst-case days — and sometimes only 4. This reduced compute time by 95%, while maintaining operator relevance.

3.6 Daily Decision Support: From Hourly to Day-Ahead Plans

We considered intertemporal modeling (i.e., optimizing switching across 24 hours), but it was too slow.

Instead, we run hourly optimizations independently and then score the results, giving higher weight to congested hours. This provides a daily plan that remains under the 20-minute runtime constraint.

4 50Hertz/Elia: High-Speed GPU-Based Topology Optimization

4.1 Presented by Nico (50Hertz)

Hi everyone. I'm Nico from 50Hertz. Together with my colleagues Leon and Christian, we've been developing a topology optimization tool that's fast enough to use in the control room — and we're now in the process of bringing it into operational use.

4.2 Objective: Speed and Scalability

The key issue in topology optimization is scale — both in terms of model size and time constraints. So our focus has been: *How can we make this process fast enough for actual day-ahead or intraday use?*

We're currently finalizing security assessments before full integration into our control center.

4.3 Tool Features and Data Sources

Our tool:

- Loads data from multiple platforms, including UCTE congestion forecasts and CGMES (we're transitioning toward CGMES-based workflows)
- Works with both Belgian and Eastern German systems — Belgian data quality is significantly better, so that's our starting point
- Now has a graphical user interface (we started in Jupyter Notebooks)

Like others, our solution is meant to be a proposal tool — it offers operators switching suggestions, not automatic enforcement.

4.4 Key Innovation: GPU-Based Genetic Algorithm

We took a radical step:

→ We run our entire optimization stack directly on GPUs.

We use:

- A custom-built DC power flow solver
- A CUDA-based genetic algorithm framework

Why?

- Matrix multiplications are the heart of both power flow and AI
- GPUs are perfect for this: fast, parallelized, and improving rapidly (e.g., Nvidia H100s)

Result: We reached 1 billion load flow computations per second, up to five orders of magnitude faster than traditional tools like Pypower.

4.5 What We Optimize

We include:

- Line switching
- Busbar reassignments
- Busbar coupler changes

Each “action” is a defined grid modification. The optimizer evaluates thousands of combinations in real time.

The genetic algorithm gives us a diverse set of near-optimal solutions, allowing us to tune aggressiveness after solving:

- Want minimal switching? Select a conservative solution.
- Want deeper congestion relief? Pick a more aggressive one.

We learned this the hard way:

Our early solutions were “too optimal” — operators refused to use them because they were too aggressive (e.g., 30+ switching actions). Now we focus on operator trust and usability.

4.6 Constraints & Realism

We’ve added:

- Limits on switching actions (e.g., 3 per hour)
- Busbar coupler logic and substation split control
- Penalties to discourage pushing problems into the distribution grid or neighboring TSOs
- Voltage angle constraints, branch loading caps, and N–1 contingencies

We even penalize switching actions that might trigger admin validation rules, even if not formally modeled.

4.7 Validation and Filtering

The optimizer runs on DC models, but we filter the results through an AC power flow after the fact:

- Invalid or risky proposals are discarded
- Visualized in UI, exported back to PowerFactory if needed

We’re not solving full OPFs yet — just topology proposals — but we approximate redispatch costs by using a proxy based on overload energy. It’s not perfect, but gives us ~70% of the insight.

4.8 Summary of Benefits

- Lightning-fast runtime (<2 minutes for optimization, <12 minutes preprocessing)
- Handles ~3,000 contingencies thanks to GPU acceleration
- Offers multiple switching options — adjustable for aggressiveness
- Integrated with operator workflows

Our biggest takeaway: Speed without trust is useless.

So we shifted focus from pure optimization to delivering credible, realistic, and tunable solutions that operators can actually use.

5 Interactive Discussion – Constraints, Trust, and Deployment Realities

5.1 Facilitators: Julian (Amprion), Christian (RWTH), and others

Participants included: PSE, 50Hertz, FGH, RWTH Aachen, and audience members (including researchers, TSOs, RCCs)

5.2 Topic 1: Real-World Constraints for Switching Actions

Julian opened with a poll using Mentimeter:

“What are the key constraints you consider when deciding switching actions?”

Key constraints surfaced:

- Maximum number of switching actions (often 3 max)
- Switching types (e.g., circuit breakers preferred over disconnectors)
- Risk and responsibility: Operators prefer actions that don’t involve high uncertainty or require on-site intervention
- Switching distance: Larger changes → higher risk (e.g., voltage violations, stability concerns)
- Asset aging and maintenance: Avoid frequent switching due to wear and maintenance costs

Nico (50Hertz): “When switching becomes aggressive, we start seeing voltage, short-circuit, or admin-rule violations we didn’t model. We now favor moderate switching — e.g., 1 or 2 substations per action.”

Energy Innovation Summit 2025 – Berlin, Day 2, June 35, 2025

Audience member: “It’s not only how many switches, but what kind. Disconnectors at night? Risky. Circuit breakers? More acceptable.”

5.3 Topic 2: Modeling Limitations and AI Use

The room discussed whether AI can support switching optimization.

Julian (Amprion): “We see AI as a possible subroutine — e.g., for narrowing the action set. But final feasibility needs to be validated by an optimization model.”

Christian (RWTH): “AI could help reduce the problem space — e.g., identifying candidate branches based on past data.”

Skepticism raised:

- AI could miss new patterns (as system configurations change)
- Past data might not represent future grid topologies
- AI does not guarantee constraint feasibility

5.4 Topic 3: Practical Considerations and Parallelization

Participants discussed performance vs. model richness.

Endika (PSE): “We accept simplifications. For us, it’s more important to have results in <45 minutes and let the operator choose from alternatives.”

Christian (RWTH): “You can run multiple approaches in parallel. A fast, heuristic model for an early suggestion, and a slower, rich model for deep evaluation.”

5.5 Topic 4: Benchmarking and Data Access

Discussion question: Should there be an open benchmark for topology optimization?

There was clear interest in a shared testbed or platform, but also acknowledgment of limitations:

- TSOs cannot freely share CGMES or UCTE data
- Researchers can’t easily access real networks with node-breaker detail
- Some attempts (e.g., via satellite data + heuristics) are promising but limited

50Hertz comment: “If anyone can create a realistic European transmission test set, with legal clearance, it would massively advance the field.”

RTE (France) and others have shared anonymized data before (e.g., via IEEE PES Challenges), but these rarely include detailed substation configurations.

6 Live Demo – Julia, PowerModels.jl, and Custom Extensions

Speaker: Tim (RWTH Aachen / Gurobi)

I’ll give you a short live demo of how we used open-source tools (Julia + PowerModels.jl) to develop a scalable SC-OTS prototype — one that works even on the German transmission grid.

6.1 Setup and Tools Used

- Language: Julia
- Base Package: PowerModels.jl
- Custom Extensions: To support:
 - Security-Constrained OTS (SC-OTS)
 - Substation reconfiguration

- Switchable branch constraints
- Export to PowerFactory and UI visualization

6.2 Demo Steps

1. Load data (IEEE 118-bus or custom German instance)
2. Solve a base AC-OPF (feasibility check)
3. Solve a DC SC-OTS with:
 - Contingency modeling
 - Connectivity constraints
 - Optional substation modeling (with switchable links)
4. Visualize switched lines (red = off)

6.3 Optimization Techniques

- Used dual-stage solving:
 - Stage 1: Security-Constrained OPF only
 - Stage 2: Allow a limited number of switching actions
- Included connectivity constraints:
 - Prevent network splits
 - Maintain generator-linking continuity
- Added post-processing filters to eliminate invalid topologies

6.4 Performance Highlights

- For the IEEE 118 test case:
 - Full SC-OTS runtime: ~5–60 seconds
 - Can add up to 3 switching actions
 - Security constraints on all contingencies
- For full German grid:
 - Runtime < 1 hour with custom heuristics
 - 2,300 buses, 40 candidate switches → solved in ~5 min

“It’s not perfect — but with some tuning and preselection, this is quite usable already.”

6.5 Closing Notes from RWTH

- The full code will be made public (after accompanying paper submission)
- Compatible with Gurobi, HiGHS, CPLEX, etc. via JuMP
- Flexible for research use and operator prototyping

7 Session Wrap-Up and Reflections

Christian (RWTH Aachen)

Thank you to all speakers and participants — I think we had an incredibly rich and practical discussion today. He emphasized a few key takeaways from the workshop:

7.1 Key Observations

1. Optimization without trust is not enough
→ Operators must trust the outcome, not just the math. Usability, explainability, and fallback options matter just as much as solution quality.
2. No single method fits all use cases
→ MILP, heuristics, generalized LODF, GPU-based GA, and AI each have trade-offs in terms of:
 - Speed
 - Constraint richness
 - Explainability
 - Scalability

3. The right “sweet spot” depends on context
→ Day-ahead vs real-time; national vs pan-European; AI-supported vs rule-based; central vs local computation — all imply different priorities.

7.2 Research and Industry Gaps

- There’s an urgent need for benchmarks and open datasets (e.g., anonymized CGMES-like models) to test and compare methods.
- Substation modeling (node-breaker detail) is often missing in shared datasets — limiting realism.
- N–1 filtering heuristics (e.g., overload proximity) are critical to reduce compute time without degrading reliability.
- Operator involvement from the start is essential — their judgment often trumps model logic.

7.3 Final Recommendations

- Do not aim for the “perfect” model — aim for something that’s trusted and used.
- Tools must offer multiple alternative solutions, not single-point optima.
- Ensure approaches are modular, so different solvers and pre-processing steps can be plugged in.
- Combine academic rigor with operational pragmatism — the summit proved this is possible.

8 Closing Julian (Amprion)

I want to thank all of you. We showed that this isn’t just a theoretical challenge — it’s a live engineering problem that multiple TSOs are actively trying to solve.

We hope to repeat this workshop format, possibly with more emphasis on benchmarking and cross-TSO collaboration.