The Sweet Spot of Bound Tightening for Topology Optimization

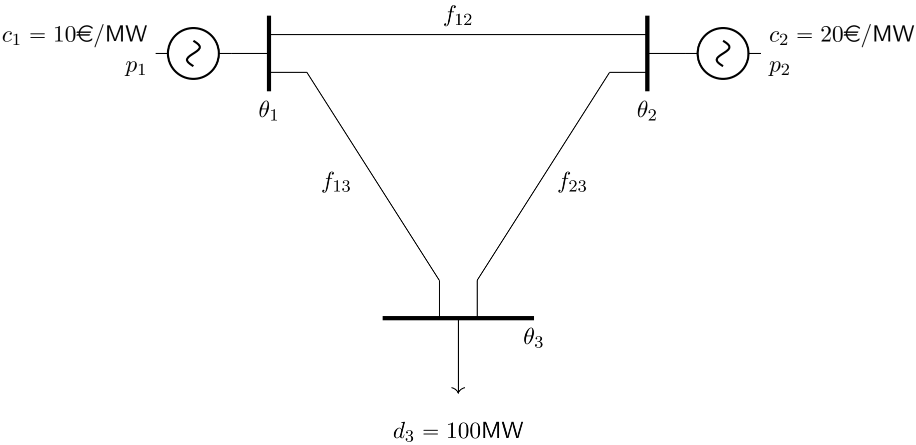
PSCC 2026 (June 9th 2026)

Salvador Pineda, Juan M. Morales

OASYS group, University of Málaga (Spain)

Motivating example

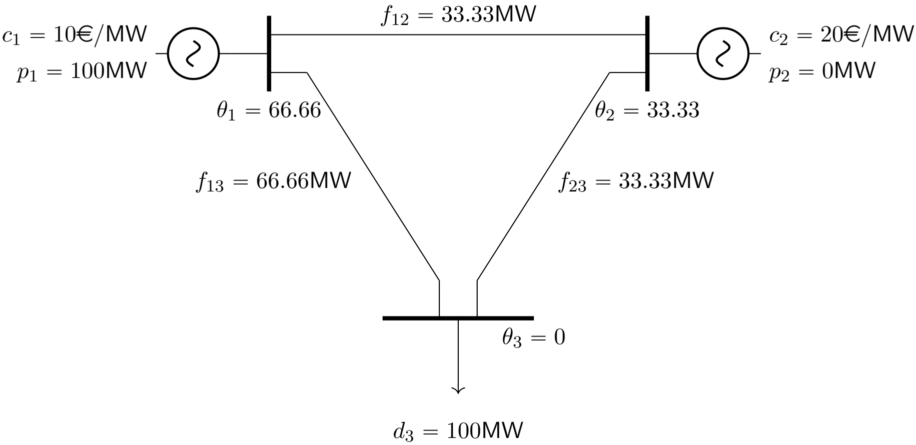
Optimal power flow (OPF): Determine the power generation and power flows to satisfy the demand at the minimum cost



The cheap unit satisfies the demand

Motivating example

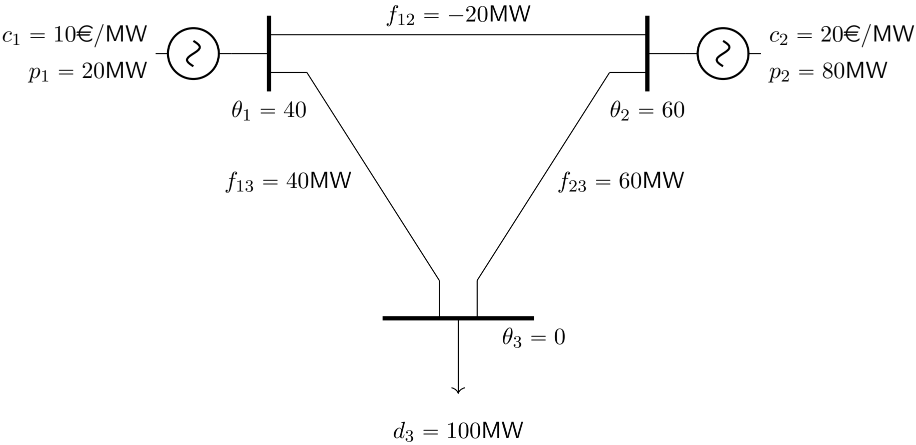
Optimal solution: generate all with cheapest unit (cost = 1000€)



Electrons are not potatoes!!!

Motivating example

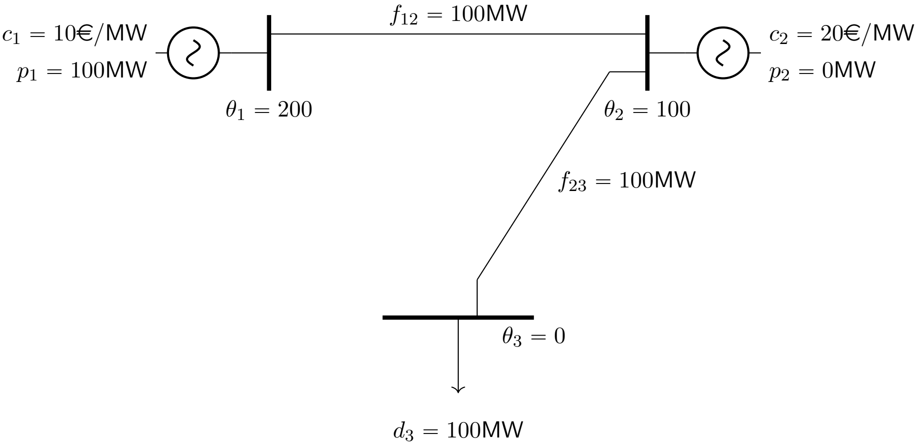
If $f_{13} \leqslant 40$, the expensive unit also generates (cost=1800€)



Network limits increase the cost!!!

Motivating example

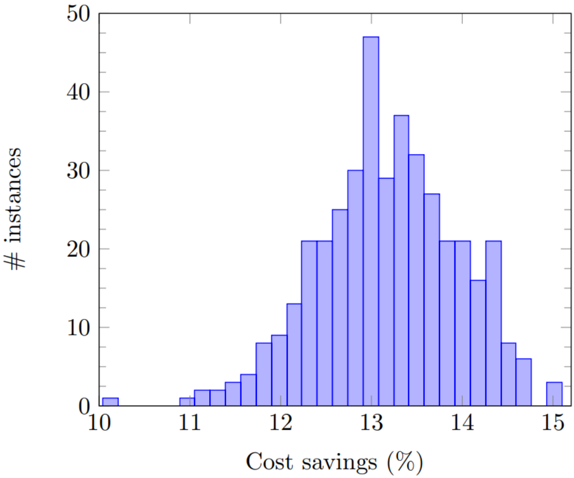
If line 13 is disconnected, cost$=$ 1000€



Disconnecting lines can reduce cost!!!

Motivating example

In the 118-bus system, the average cost saving is 13.2%



Methodology

The optimal power flow (OPF) is formulated as a LP problem as follows

$$\min_{p_n, f_l, \theta_n} \quad \sum_n c_n \, p_n \tag{1a}$$

$$\text{s.t.} \quad \sum_{l \in \mathcal{L}(n,\cdot)} f_l - \sum_{(l) \in \mathcal{L}(\cdot,n)} f_l = p_n - d_n, \quad \forall n \tag{1b}$$

$$f_l = b_l(\theta_n - \theta_m), \quad \forall l = (n, m) \in \mathcal{L} \tag{1c}$$

$$\underline{p}_n \leqslant p_n \leqslant \overline{p}_n, \quad \forall n \tag{1d}$$

$$-\underline{f}_l \leqslant f_l \leqslant \overline{f}_l, \quad \forall l \in \mathcal{L} \tag{1e}$$

The optimal transmission switching (OTS) requires dummy flow variable $\tilde{f}_l$ and binary variable x_l and is formulated as a mixed-integer non-linear problem ...

$$
\begin{aligned}
\min_{p_n, f_l, \theta_n, x_l} \quad & \sum_n c_n \, p_n && \text{(2a)} \\
\text{s.t.} \quad & \sum_{l \in \mathcal{L}(n,\cdot)} f_l - \sum_{l \in \mathcal{L}(\cdot,n)} f_l = p_n - d_n, \quad \forall n && \text{(2b)} \\
& \tilde{f}_l = b_l(\theta_n - \theta_m), \quad \forall l = (n,m) \in \mathcal{L} && \text{(2c)} \\
& f_l = x_l \tilde{f}_l, \quad \forall l \in \mathcal{L} && \text{(2d)} \\
& \underline{p}_n \leqslant p_n \leqslant \overline{p}_n, \quad \forall n && \text{(2e)} \\
& -x_l \underline{f}_l \leqslant f_l \leqslant x_l \overline{f}_l, \quad \forall l \in \mathcal{L} && \text{(2f)} \\
& x_l \in \{0, 1\}, \quad \forall l \in \mathcal{L} && \text{(2g)}
\end{aligned}
$$

... that can be directly solved using optimization solvers such as Gurobi.

Methodology

To avoid the non-linear terms in

$$f_l = x_l \tilde{f}_l$$

We replace it by

$$\underline{M}_l(1 - x_l) \leqslant -f_l + \tilde{f}_l \leqslant \overline{M}_l(1 - x_l)$$

Together with equation

$$-x_l \underline{f}_l \leqslant f_l \leqslant x_l \overline{f}_l$$

We have that:

- If $x_l = 1 \Rightarrow 0 \leqslant -f_l + \tilde{f}_l \leqslant 0$ and $-\underline{f}_l \leqslant f_l \leqslant \overline{f}_l$

- If $x_l = 0 \Rightarrow f_l = 0$ and $\underline{M}_l \leqslant \tilde{f}_l \leqslant \overline{M}_l$

- If $x_l = 0.5 \Rightarrow 0.5\underline{M}_l \leqslant -f_l + \tilde{f}_l \leqslant 0.5\overline{M}_l$ and $-0.5\underline{f}_l \leqslant f_l \leqslant 0.5\overline{f}_l$

Fractional values of x_l lead to pipeline model with reduced capacities!!

The OTS is reformulated as a mixed-integer linear problem

$$\min_{p_n, f_l, \theta_n, x_l} \quad \sum_n c_n\, p_n \tag{3a}$$

$$\text{s.t.} \quad \sum_{l \in \mathcal{L}(n,\cdot)} f_l - \sum_{l \in \mathcal{L}(\cdot,n)} f_l = p_n - d_n, \quad \forall n \tag{3b}$$

$$\tilde{f}_l = b_l(\theta_n - \theta_m), \quad \forall l = (n,m) \in \mathcal{L} \tag{3c}$$

$$\underline{M}_l(1 - x_l) \leqslant -f_l + \tilde{f}_l \leqslant \overline{M}_l(1 - x_l), \quad \forall l \in \mathcal{L} \tag{3d}$$

$$\underline{p}_n \leqslant p_n \leqslant \overline{p}_n, \quad \forall n \tag{3e}$$

$$-x_l \underline{f}_l \leqslant f_l \leqslant x_l \overline{f}_l, \quad \forall l \in \mathcal{L} \tag{3f}$$

$$x_l \in \{0,1\}, \quad \forall l \in \mathcal{L} \tag{3g}$$

$\underline{M}_l$ and $\overline{M}_l$ must be large enough to be valid bounds for $\tilde{f}_l$ if line is open

$\underline{M}_l$ and $\overline{M}_l$ must be small enough to avoid computational issues

Optimization-based bound tightening

$\mathcal{R}(\mathbf{F},\mathbf{M})$: Feasible region (3b)-(3f) for line capacities $\mathbf{F}$ and big-Ms $\mathbf{M}$.
$\mathcal{X}^B := \{\mathbf{x} \in \{0,1\}^{|\mathcal{L}|}\}$ Exact binary switching domain.
$\mathcal{X}^R := \{\mathbf{x} \in [0,1]^{|\mathcal{L}|}\}$ Continuous binary relaxation.
$\mathcal{C} := \{\mathbf{p} \in \mathbb{R}^{|\mathcal{N}|} : \sum_n c_n p_n \leqslant \overline{C}\}$ Cost budget constraint.

Bounding problems (MIPs): Tight bounds but difficult to solve

$$\underline{f}_l \,/\, \overline{f}_l = \min_{\mathcal{R}(\mathbf{F},\mathbf{M}) \cap \mathcal{X}^B \cap \mathcal{C}} / \max \quad f_l \quad \text{s.t.} \quad x_l = 1 \tag{4a}$$

$$\underline{M}_l \,/\, \overline{M}_l = \min_{\mathcal{R}(\mathbf{F},\mathbf{M}) \cap \mathcal{X}^B \cap \mathcal{C}} / \max \quad \tilde{f}_l \quad \text{s.t.} \quad x_l = 0 \tag{4b}$$

Bounding problems (LPs): Loose bounds but easy to solve

$$\underline{f}_l \,/\, \overline{f}_l = \min_{\mathcal{R}(\mathbf{F},\mathbf{M}) \cap \mathcal{X}^R \cap \mathcal{C}} / \max \quad f_l \quad \text{s.t.} \quad x_l = 1 \tag{5a}$$

$$\underline{M}_l \,/\, \overline{M}_l = \min_{\mathcal{R}(\mathbf{F},\mathbf{M}) \cap \mathcal{X}^R \cap \mathcal{C}} / \max \quad \tilde{f}_l \quad \text{s.t.} \quad x_l = 0 \tag{5b}$$

Research question

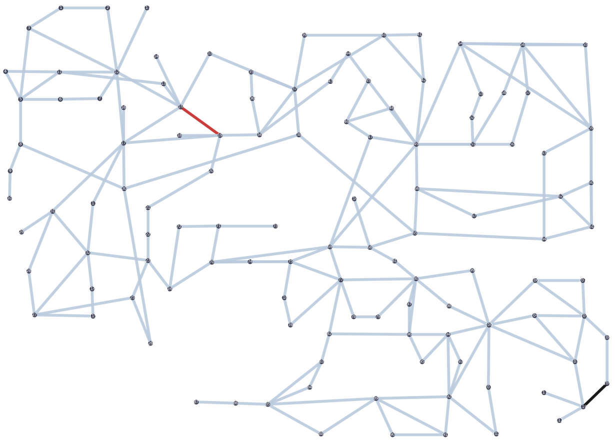
Does it make sense to solve bounding problems with a subset of the switching variables as binary while the rest are relaxed?

What we know:

- High # binaries: tighter bounds but difficult to solve
- Low # binaries: looser bounds but easy to solve
- Relaxed switching allows for flows that violate physics laws

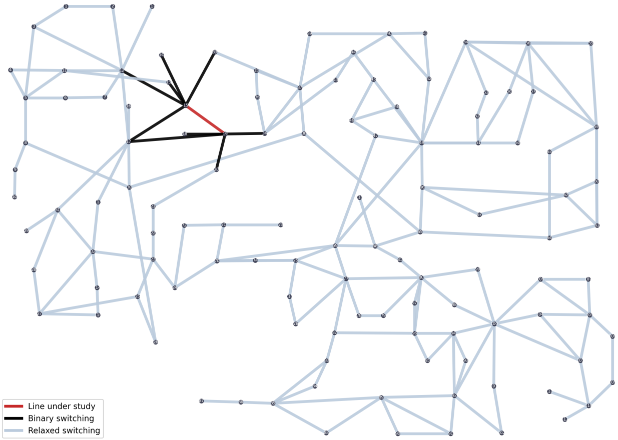
The idea

Lines topologically distant from a target line have a low impact on its bounds; relaxing them saves time with negligible accuracy loss.



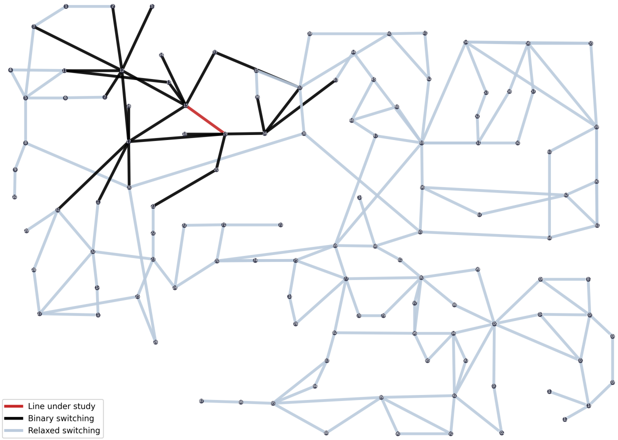
The idea

Retain binary decisions for lines directly connected to the target line, and continuously relax the switching variables for rest of the network (level 1)



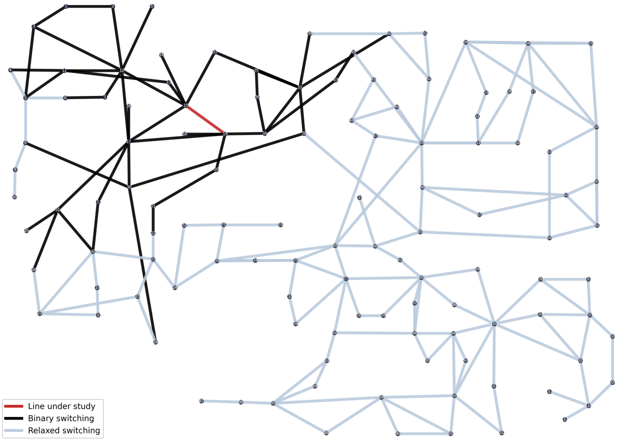
The idea

Extend the binary region to include the Level 1 lines and all lines directly connected to them (level 2)



The idea

Further expand the binary region to include the Level 2 lines and all lines directly connected to them (level 3)



Methodology

- **Topological Neighborhoods ($\mathcal{L}_l^k$):** For a target line $l \in \mathcal{L}$, closeness $k \in \mathbb{N}$, and $\mathcal{L}_l^0 := \varnothing$, the sets are defined recursively:
  - **Level-1 ($\mathcal{L}_l^1$):** All lines sharing at least one terminal bus with line l, excluding line l itself.
  - **Level-k ($\mathcal{L}_l^k$):** The union of $\mathcal{L}_l^{k-1}$ and all lines sharing at least one terminal bus with any line in $\mathcal{L}_l^{k-1}$, satisfying the nesting property:

$$\mathcal{L}_l^1 \subseteq \mathcal{L}_l^2 \subseteq \cdots \subseteq \mathcal{L}_l^k.$$

- **Partial Binary Relaxation Space ($\mathcal{X}_l^k$):** The space of line status vectors where only variables within topological distance k are constrained to be binary, while the remaining variables are relaxed:

$$\mathcal{X}_l^k := \left\{ \mathbf{x} \in [0,1]^{|\mathcal{L}|} : x_j \in \{0,1\} \ \forall j \in \mathcal{L}_l^k \right\}$$

Methodology

- Using the partially relaxed feasible set $\mathcal{X}_l^k$, the bounding problems are defined as follows:

$$\underline{f}_l \,/\, \overline{f}_l = \min_{\mathcal{R}(\mathbf{F},\mathbf{M}) \cap \mathcal{X}_l^k \cap \mathcal{C}} / \max \quad f_l \quad \text{s.t.} \quad x_l = 1 \tag{6a}$$

$$\underline{M}_l \,/\, \overline{M}_l = \min_{\mathcal{R}(\mathbf{F},\mathbf{M}) \cap \mathcal{X}_l^k \cap \mathcal{C}} / \max \quad \tilde{f}_l \quad \text{s.t.} \quad x_l = 0 \tag{6b}$$

- Enforcing integrality near line l ensures Kirchhoff's laws are satisfied locally, preventing overly loose bounds.
- Distant lines are relaxed to reduce computational burden, creating a controllable trade-off between speed and bound quality.
- We denote the proposed approach as `TBT-`k (*Topological Bound Tightening*).

Case study

- IEEE 118-bus network (all lines switchable)
- 300 instances with random demand ($\pm 10\%$ variation)
- Gurobi 10.0.3 with 0.01% gap and maximum time 1h.
- Benchmark approaches:
  - `MIP`: Solves OTS as a mixed-integer linear program using the original, unrefined big-M bounds.
  - `BLP`: Solves OTS as a mixed-integer bilinear program; the relation $f_l = x_l \tilde{f}_l$ is handled natively by the solver.
  - `IND`: Replaces the bilinear relation with indicator constraints, avoiding big-M constants.
  - `SBT`-t (*Solver Bound Tightening*): Computes bounds by keeping **all** variables binary ($\mathcal{X}^B$) with a strict time limit of t milliseconds per problem, returning the best bound found.

Results

	Bound Red. (%)		Runtime (s)			Timeouts
	ΔF	ΔM	Bounding	OTS	Total	#TO
TBT-0	10.11	0.00	8.78	298.89	307.67	19
TBT-1	12.57	1.50	12.77	257.94	270.71	16
TBT-2	14.23	3.44	20.80	159.88	180.68	9
TBT-3	17.91	4.82	42.18	237.16	279.34	13
TBT-4	20.50	5.89	145.33	133.67	279.00	8
TBT-5	21.08	10.00	384.45	136.56	521.01	6

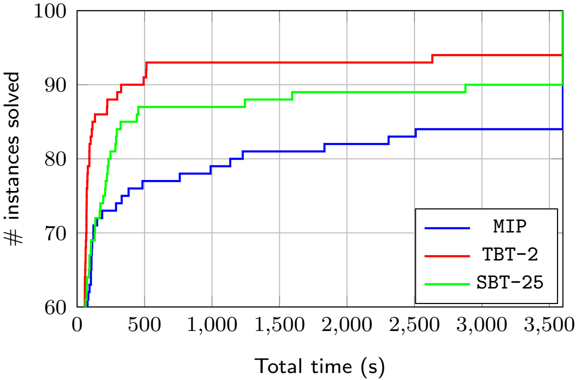
- Increasing the topological distance k tightens the bounds at the expense of higher computational time per bounding problem.
- The number of instances unsolved within the 1-hour limit generally tends to decrease as k increases.
- The topological distance $k = 2$ serves as the sweet spot that achieves the lowest total runtime with a low number of unsolved instances.

Results

	Bound Red. (%)		Runtime (s)			Timeouts
	ΔF	ΔM	Bounding	OTS	Total	#TO
MIP	-	-	-	327.22	327.22	20
IND	-	-	-	1026.21	1026.21	60
BLP	-	-	-	2233.53	2233.53	160
TBT-2	14.23	3.44	20.80	159.88	180.68	9
SBT-25	11.81	2.09	18.29	216.83	235.12	11

- IND and BLP are highly inefficient, suffering from extremely high runtimes and a large number of unsolved instances.
- TBT-2 and SBT-25 substantially outperform MIP in both total computation time and overall completion rates.
- TBT-2 is the most efficient approach, yielding a 1.8x speedup over MIP and a 1.3x speedup over SBT-25.

Results



- Evaluating the subset of the 100 hardest instances highlights a much wider performance gap between the approaches.
- On these challenging instances, `TBT-2` dominates by achieving a 2.3x speedup over `MIP` and a 1.6x speedup over `SBT-25`.
- `TBT-2` minimizes failures on hard instances, leaving only 6 unsolved compared to 10 for `SBT-25` and 16 for `MIP`.

Conclusions

- **Topology-Aware Tightening:** Selective binary constraints near target lines preserve local physical laws, preventing loose relaxations.
- **Level-2 Sweet Spot:** Closeness level $k = 2$ maximizes bound tightness while maintaining computational tractability.
- **Overall Performance (300 Instances):** `TBT-2` emerges as the most efficient approach, yielding a 1.8x speedup over `MIP` and a 1.3x speedup over `SBT-25`.
- **Stress Test (100 Hardest Instances):** `TBT-2` dominates challenging scenarios with a 2.3x speedup over `MIP` (1.6x over `SBT-25`), leaving only 6 unsolved instances compared to 16 for `MIP`.

A Final Reflection

- Learning-based approaches for topology optimization thoroughly detail training parameters (epochs, learning rates, decay rates, batch sizes) that often require weeks of extensive tuning.

- Yet, when benchmarking against conventional optimization, the tuning of critical big-M constants is rarely mentioned or performed, despite heavily impacting solver runtime.

- If weeks are spent optimizing machine learning hyperparameters, at least a few hours must be dedicated to tuning baseline optimization parameters like big-M constants for topology optimization.

Thanks for the attention!!

Questions??

OASYS
Optimization and Analytics for Sustainable energY Systems

More info: oasys.uma.es
Email: spineda@uma.es