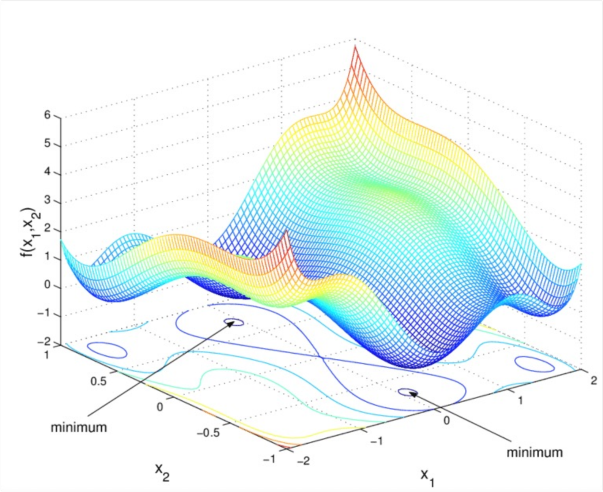
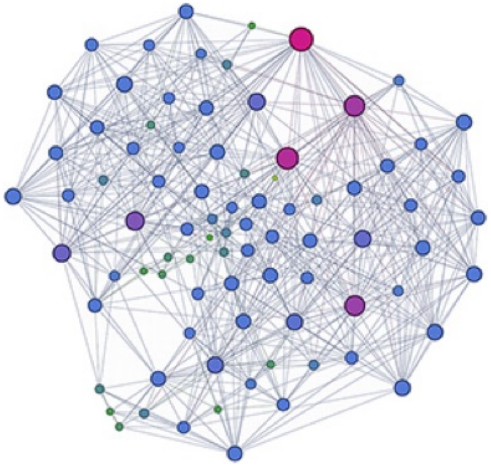
GravityX

Hassan Hijazi

ARPA-E Grid Optimization Competition





Challenge 3: The Hardest Challenge of them All

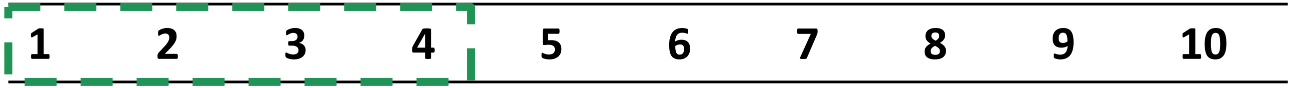
Main Challenges:

- Temporal Constraints (18 to 48 time-steps)
- 1e-8 Constraint Satisfaction (4 orders of magnitude drop!)
- Dense Reserve Constraints (thousands of nnz in one constraint)
- Different N-1 Post-Contingency Model

2019-2020 2020-2021 2021-2022 2022-2023

GravityX's Approach

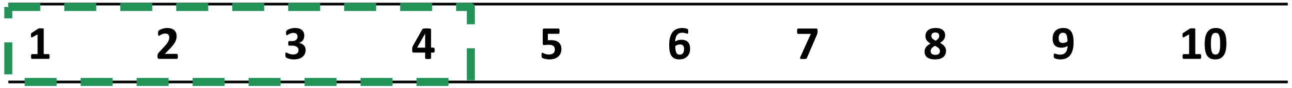
Decomposition + MIP + NLP



Rolling Horizon Time Decomposition

GravityX's Approach

Decomposition + MIP + NLP



Rolling Horizon Time Decomposition

In hindsight, a full-horizon copper-plate (no flow vars) model as a starting point was important

GravityX's Approach

Decomposition + MIP + NLP

1	2	3	4	5	6	7	8	9	10

$$\left\{\begin{array}{rl} \min & \boldsymbol{c_1}^t x + \boldsymbol{c_2}^t y \\ s.t. & \boldsymbol{A}_1 x + \boldsymbol{A}_2 y \leq \boldsymbol{b} \\ & x \in \mathbb{R}^n, y \in \mathbb{Z}^n \end{array}\right\}$$

GUROBI OPTIMIZATION

From Lossless to Lossy Mixed-Integer Linear Power Flow Model (including reactive power)

GravityX's Approach

Algorithm 2 MIP-NLP Decomposition with Backtracking

```
Fix UC binaries to the previous operating point solution.
Solve resulting ACOPF and compute line losses.
Fix active and reactive line losses using AC solution
Decompose UC MIP & solve using rolling-horizon.
while not fix point or time limit reached do
    Fix UC binaries to the previous MIP solution.
    Solve resulting ACOPF and compute line losses.
    Fix active and reactive line losses using AC solution
    Decompose UC MIP & solve using rolling-horizon.
    if infeasible MIP then backtrack the initial time step
    end if
end while
while not time limit reached do
    Fix UC binaries to the optimal MIP solution.
    Decompose NLP & solve using rolling-horizon.
    if infeasible NLP then backtrack the initial time step
    end if
end while
```

New Contributions

Valid Linear Constraints (Exact)

$$\boldsymbol{g}_e(p_e^{fr} - p_e^{to}) - \boldsymbol{b}_e(q_e^{fr} - q_e^{to}) =$$

$$\boldsymbol{g_e}\left(\boldsymbol{g_e} + \boldsymbol{g_e^{fr}}\right)\frac{w_i}{\boldsymbol{\tau_e^2}} - \boldsymbol{g_e}\left(\boldsymbol{g_e} + \boldsymbol{g_e^{to}}\right)w_j +$$

$$\boldsymbol{b}_e\left(\boldsymbol{b}_e + \boldsymbol{b}_e^{fr} + \frac{\boldsymbol{b}_e^{ch}}{2}\right)\frac{w_i}{\boldsymbol{\tau_e^2}} - \boldsymbol{b}_e\left(\boldsymbol{b}_e + \boldsymbol{b}_e^{to} + \frac{\boldsymbol{b}_e^{ch}}{2}\right)w_j$$

New Contributions

Transformer Power Flow Reformulation

$$p_e^{fr} = \left(\boldsymbol{g}_e + \boldsymbol{g}_{\boldsymbol{e}}^{\boldsymbol{fr}}\right) \frac{v_i^2}{\tau_e^2} - \frac{v_i v_j}{\tau_e} \left(\boldsymbol{g}_e \cos(\theta_i - \theta_j - \theta_e) + \boldsymbol{b}_e \sin(\theta_i - \theta_j - \theta_e)\right)$$

$$p_e^{to} = \left(\boldsymbol{g}_{\boldsymbol{e}} + \boldsymbol{g}_{\boldsymbol{e}}^{\boldsymbol{to}}\right) v_j^2 - \frac{v_i v_j}{\tau_e} \left(\boldsymbol{g}_e \cos(\theta_j - \theta_i + \theta_e) + \boldsymbol{b}_e \sin(\theta_j - \theta_i + \theta_e)\right)$$

$$q_e^{fr} = -\left(\boldsymbol{b}_e + \boldsymbol{b}_e^{fr} + \frac{\boldsymbol{b}_e^{ch}}{2}\right) \frac{v_i^2}{\tau_e^2} + \frac{v_i v_j}{\tau_e} \left(\boldsymbol{b}_e \cos(\theta_i - \theta_j - \theta_e) - \boldsymbol{g}_e \sin(\theta_i - \theta_j - \theta_e)\right)$$

$$q_e^{to} = -\left(\boldsymbol{b}_e + \boldsymbol{b}_e^{to} + \frac{\boldsymbol{b}_e^{ch}}{2}\right) v_j^2 + \frac{v_i v_j}{\tau_e} \left(\boldsymbol{b}_e \cos(\theta_j - \theta_i + \theta_e) - \boldsymbol{g}_e \sin(\theta_j - \theta_i + \theta_e)\right)$$

New Contributions

Transformer Power Flow Reformulation

$$p_e^{fr} = \left(\boldsymbol{g_e} + \boldsymbol{g_e^{fr}}\right)\frac{v_i^2}{\tau_e^2} - \frac{v_i v_j}{\tau_e}\left(\boldsymbol{g}_e \cos(\theta_i - \theta_j - \theta_e) + \boldsymbol{b}_e \sin(\theta_i - \theta_j - \theta_e)\right)$$

$$p_e^{to} = \left(\boldsymbol{g_e} + \boldsymbol{g_e^{to}}\right) v_j^2 - \frac{v_i v_j}{\tau_e}\left(\boldsymbol{g}_e \cos(\theta_j - \theta_i + \theta_e) + \boldsymbol{b}_e \sin(\theta_j - \theta_i + \theta_e)\right)$$

$$q_e^{fr} = -\left(\boldsymbol{b}_e + \boldsymbol{b}_e^{fr} + \frac{\boldsymbol{b}_e^{ch}}{2}\right)\frac{v_i^2}{\tau_e^2} + \frac{v_i v_j}{\tau_e}\left(\boldsymbol{b}_e \cos(\theta_i - \theta_j - \theta_e) - \boldsymbol{g}_e \sin(\theta_i - \theta_j - \theta_e)\right)$$

$$q_e^{to} = -\left(\boldsymbol{b}_e + \boldsymbol{b}_e^{to} + \frac{\boldsymbol{b}_e^{ch}}{2}\right) v_j^2 + \frac{v_i v_j}{\tau_e}\left(\boldsymbol{b}_e \cos(\theta_j - \theta_i + \theta_e) - \boldsymbol{g}_e \sin(\theta_j - \theta_i + \theta_e)\right)$$

$$\tau_e^{'}\tau_e^2 = 1$$

$$p_e^{fr} = \left(\boldsymbol{g_e} + \boldsymbol{g_e^{fr}}\right) v_i^2 \tau_e^{'} - v_i v_j \tau_e^{'} \tau_e \left(\boldsymbol{g}_e \cos(\theta_i - \theta_j - \theta_e) + \boldsymbol{b}_e \sin(\theta_i - \theta_j - \theta_e)\right)$$

$$p_e^{to} = \left(\boldsymbol{g_e} + \boldsymbol{g_e^{to}}\right) v_j^2 - v_i v_j \tau_e^{'} \tau_e \left(\boldsymbol{g}_e \cos(\theta_j - \theta_i + \theta_e) + \boldsymbol{b}_e \sin(\theta_j - \theta_i + \theta_e)\right)$$

$$q_e^{fr} = -\left(\boldsymbol{b}_e + \boldsymbol{b}_e^{fr} + \frac{\boldsymbol{b}_e^{ch}}{2}\right) v_i^2 \tau_e^{'} + v_i v_j \tau_e^{'} \tau_e \left(\boldsymbol{b}_e \cos(\theta_i - \theta_j - \theta_e) - \boldsymbol{g}_e \sin(\theta_i - \theta_j - \theta_e)\right)$$

$$q_e^{to} = -\left(\boldsymbol{b}_e + \boldsymbol{b}_e^{to} + \frac{\boldsymbol{b}_e^{ch}}{2}\right) v_j^2 + v_i v_j \tau_e^{'} \tau_e \left(\boldsymbol{b}_e \cos(\theta_j - \theta_i + \theta_e) - \boldsymbol{g}_e \sin(\theta_j - \theta_i + \theta_e)\right)$$

NLP Reformulation
Does it make a difference?

Instance	Scenario	Original	Reformulation
C3E3N01576D1	scenario_027	14 sec	8 sec
C3E3N04224D1	scenario_131	14 sec	10 sec
C3E3N06049D1	scenario_031	20 sec	14 sec
C3E3N06717D1	scenario_031	>145 sec	26 sec
C3E3N08316D1	scenario_001	>145 sec	33 sec

New Contributions

Linear AC Losses

$$p_{eij} + p_{eji} = \boldsymbol{pl_e}$$

$$q_{eij} + q_{eji} = \boldsymbol{ql_e}$$

Algorithm 2 MIP-NLP Decomposition with Backtracking

Fix UC binaries to the previous operating point solution.
Solve resulting ACOPF and compute line losses.
Fix active and reactive line losses using AC solution
Decompose UC MIP & solve using rolling-horizon.
while not fix point or time limit reached **do**
    Fix UC binaries to the previous MIP solution.
    Solve resulting ACOPF and compute line losses.
    Fix active and reactive line losses using AC solution
    Decompose UC MIP & solve using rolling-horizon.
    if infeasible MIP **then** backtrack the initial time step
    end if
end while
while not time limit reached **do**
    Fix UC binaries to the optimal MIP solution.
    Decompose NLP & solve using rolling-horizon.
    if infeasible NLP **then** backtrack the initial time step
    end if
end while

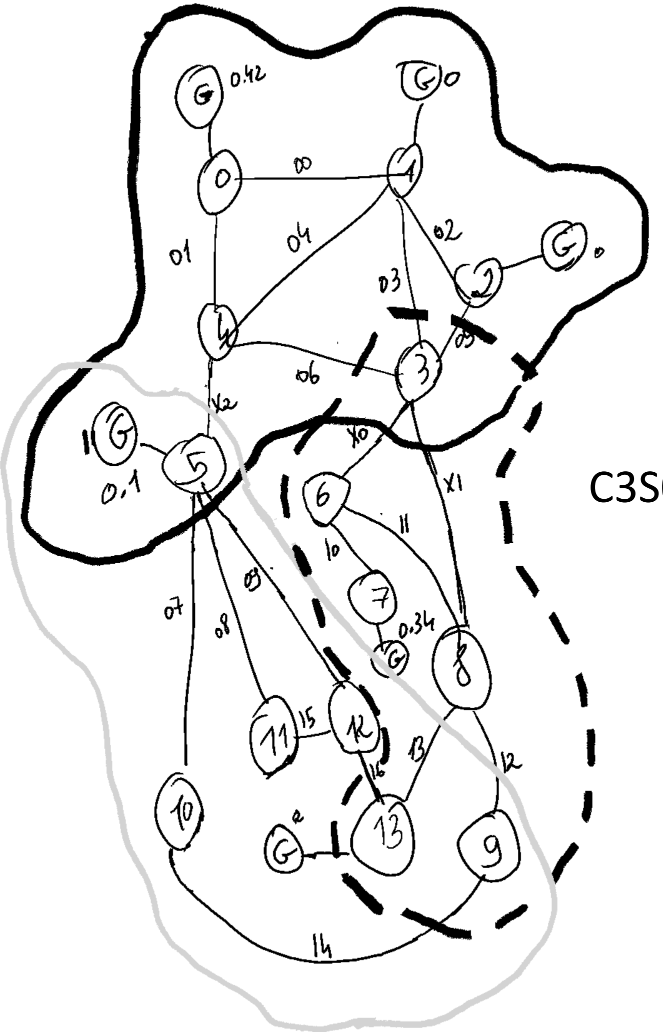
Linear AC Losses

Does it make a difference?

Instance	Scenario	Lossless-NF	Lossless-DC	Lossy-DC	LAC-Losses
C3E3N01576D1	scenario_027	96,227,154	98,057,070	97,780,021	98,298,517
C3E3N04224D1	scenario_131	91,168,493	91,141,339	91,142,009	91,169,130
C3E3N06049D1	scenario_031	104,152,318	104,095,683	104,093,510	104,135,871
C3E3N06717D1	scenario_031	Fail	-889,688,701	136,130,464	136,589,493
C3E3N08316D1	scenario_001	1,158,730,927	Fail	Fail	1,180,957,975

Things I started but did not have time to finish:

1) Spatial Decomposition

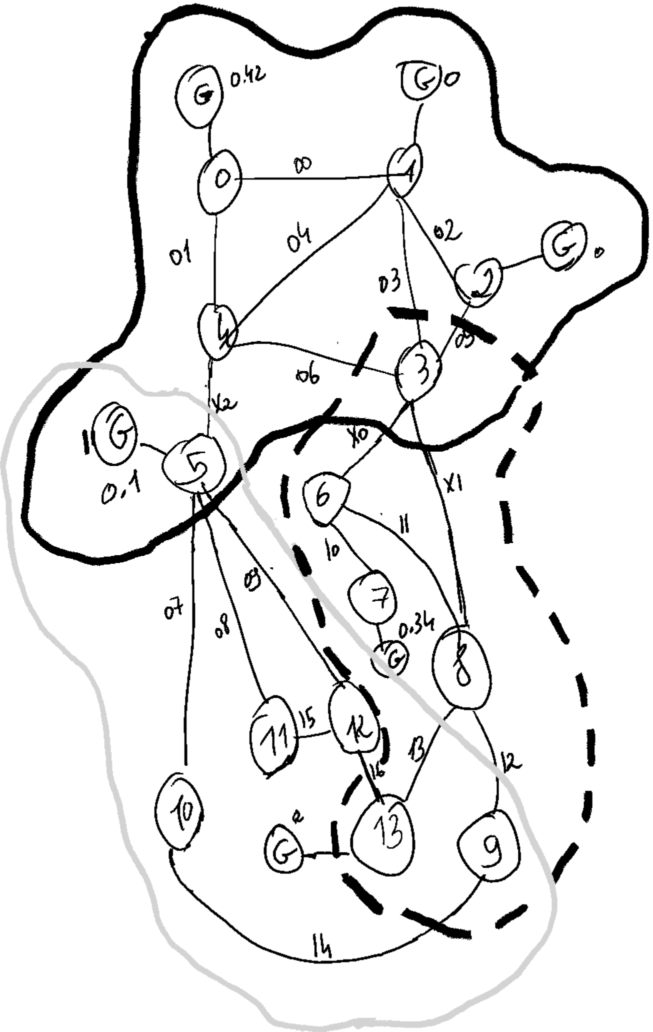


C3S0N00014

Things I started but did not have time to finish:

1) Spatial Decomposition

2) Dynamic Contingency Constraint Generation

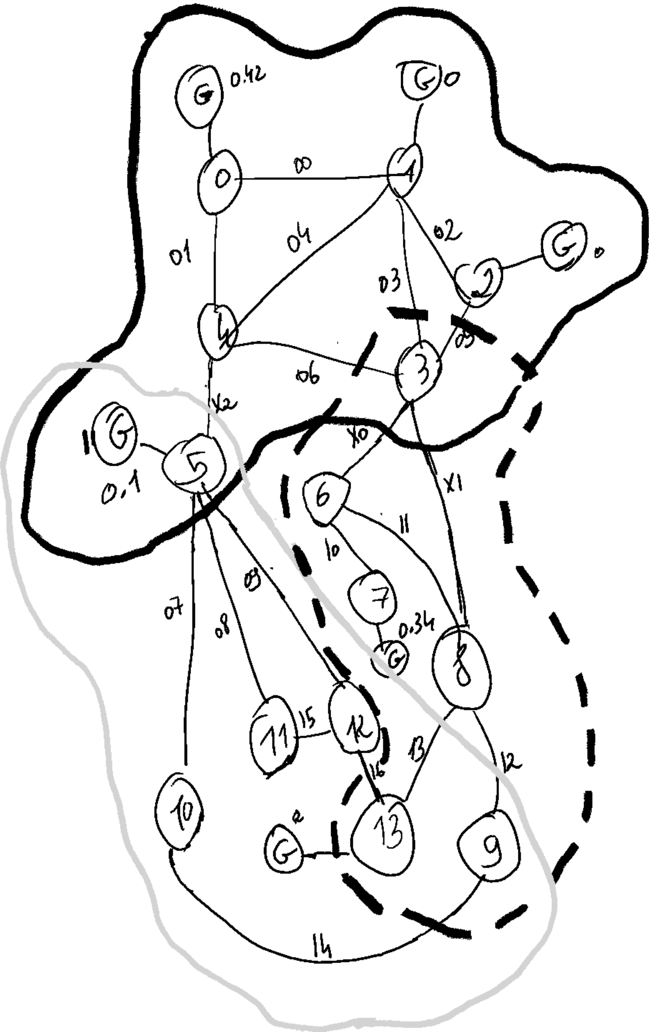


Things I started but did not have time to finish:

1) Spatial Decomposition

2) Dynamic Contingency Constraint Generation

3) Project the slack variables and use on/off constraints for line switching



1. What were the 10 most difficult scenarios to solve? And why?
    - The 23k instance was the hardest, Ipopt convergence on a single time step was challenging, spatial decomposition was required
2. What were the 10 least difficult scenarios to solve? And why?
    - The small instances up to 1500 were easy to solve since all models were converging fast enough
3. Did you find difficulties with industrial networks (not released networks)? Explain why?
    - GravityX's algorithm failed on some Event 4 Industrial scenarios due to the temporal decomposition leading to infeasible points, could have been mitigated by using copper-plate MIP on full horizon

7. Which constraints were more challenging to satisfy?
    - Reserve and N-1 constraints, former due to density, latter due to size
8. How would you compare the computational complexity of larger grids with the small grids?
    - There is a step-change going from 8k to 23k
9. Regarding the data input format, is it easier to parse?
    - Yes
10. What are the main differences in the optimization behavior of D1, D2 and D3?
    - D1 was the hardest due to tight wall-clock time constraints

1. What process did your team use in deciding the algorithmic approach?
    - Start with full model, simplify/relax/reformulate/decompose until time limit is satisfied
2. Did your team consider/use a hybrid approach by running different types of algorithms in parallel?
    - The same algorithm was used but with different parametrization (e.g., excluding reserve constraints, different time horizon, with/without MIP, etc..)
3. Did you/your team consider adjusting the parameters/heuristics of your algorithm based on network characteristics? If yes, explain how?
    - Time decomposition horizon was a function of network size, a threshold was also set for adding/excluding contingency constraints
4. Did you/your team try to use any machine learning approach to learn the Sandbox datasets?
    - No
5. Did you/your team consider changing the algorithmic approach/modeling approach when new datasets are published? If yes, why?
    - Yes, I started implementing a spatial decomposition method for the 23K instance but ran out of time

6. Did the teams consider a "simultaneous multi-period" OPF approach (as opposed to considering each time period individually)? If so, how did it scale and what, if any, were the benefits to solution quality?
    - Yes, a multi-period ACOPF model was scalable up to 2k networks, a few percent improvement in the objective value.
7. How (if at all) did your team incorporate reserve constraints into the OPF subproblem(s)?
    - Added as is.
8. UC determines the binary variables and some continuous variables. We understand fixing the binary variables makes the remaining AC OPF a continuous nonconvex programming. How do you treat the decision of the continuous variables determined by UC?
    - Ignored all continuous variables from UC.
9. How do you update UC decisions if you find the first UC is not optimal or feasible?
    - I included backtracking in my temporal decomposition.

Thanks!

”Competition is a lot like cod liver oil. First it makes you sick. Then it makes you better.” - Unattributed

“If you can't win, make the fellow ahead of you break the record.” - EVAN ESAR

“It is in vain for us to devise schemes by which competition can be put out of civilized life. Competition is the condition of life.” - LYMAN ABBOTT

“The ultimate victory in competition is derived from the inner satisfaction of knowing that you have done your best and that you have gotten the most out of what you had to give” - Howard Cosell