WASHINGTON STATE
UNIVERSITY

# Scalable Multi-Period Optimal Power Flow for Active Power Distribution Systems

*or simply, Scalable MP-OPF in ADS*

*Aryan Ritwajeet Jha*

*Graduate Research Assistant*

*Washington State University*

# **Steps towards understanding Temporal Decomposition** – Which Papers to Read?

- Find work where it has actually been used.
  - Beginning with only those which solve for the MPOPF problem.
    - Right now, Transmission systems are OK too (they have to deal with ramping up/down constraints for generators)
    - But may look into other domains as well afterwards
  - Ascertaining which approach to follow, based on certain desirable features the *future algorithm* should have.

# **Steps towards understanding Temporal Decomposition** – Intended Algorithmic Solution

- Desirable features the *future algorithm* should have:
  1. Should be **scalable with** respect to **Time** (horizon duration)
     - At least, something like 96 time-steps should be solvable in one go
  2. Should ideally **solve** for **the** 'true' **nonlinear problem**, with as few approximations as possible
  3. **Relaxations** however **are fine**, and if they're used, their solutions will obviously be checked for any infeasibility.
  4. Changing **optimization formulation** is undesirable for me. Something **close to**, if not exactly the same as **the 'standard' nonlinear Branch Flow Model** [farivar2013] is what I want.
     - So, I'm looking to solve the problem using Branch Flow Variables $P_{ij}, Q_{ij}, v_j, l_{ij}$ only.
     - *Many* papers, bring out their own new set of variables, approximations, customized algorithms to deal with them, in order to solve for the MPOPF problem. This is NOT what I'm looking to do.

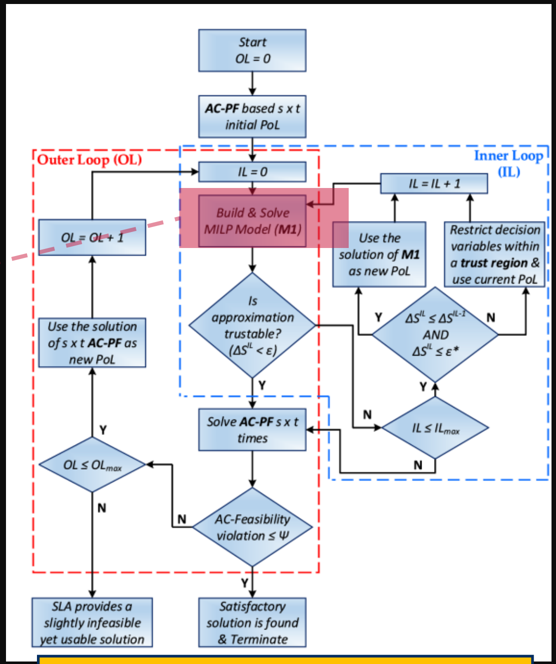
## **Do others 'scalably' solve MPOPF?** – Some of them, Yes.

- These days, many papers pose a solution to the MPOPF problem.
  - Many papers don't actually solve the problem in a scalable manner: the test-cases are reasonable (moderate sized, moderate penetration, day-ahead or large sized, low penetration, some hours only), but even they acknowledge that this formulation is not scalable, unless …
  - … Approximations are made

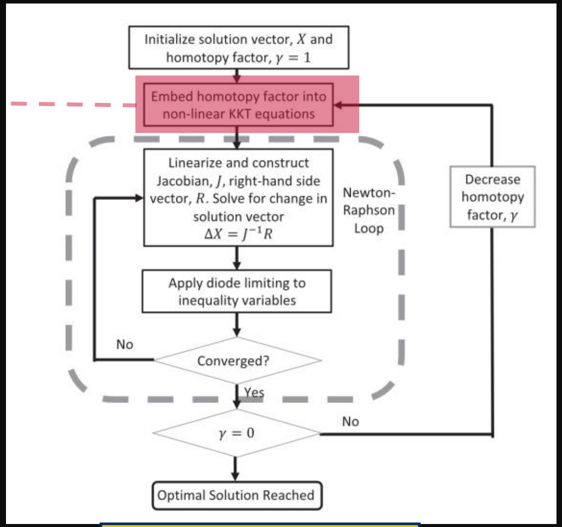
# How do others 'scalably' solve it? – Linearized Solution, Iteratively Improved

Specifically,

- … Approximations are made at the most *modular form of OPF formulation* [usman2022, usman2023] which itself could be within a loop, where some meta-parameters are adjusted based on successive iterative updates.
  - Some papers model these successive iterative updates on a homotopy based paradigm [agarwal2022]




MILP call inside a bigger algorithm

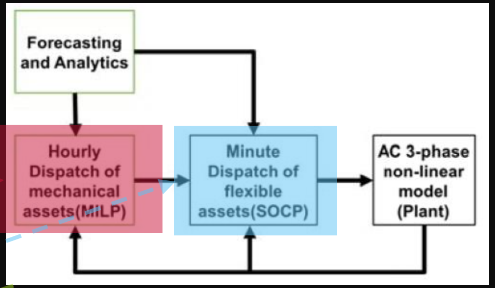



Some homotopy thing

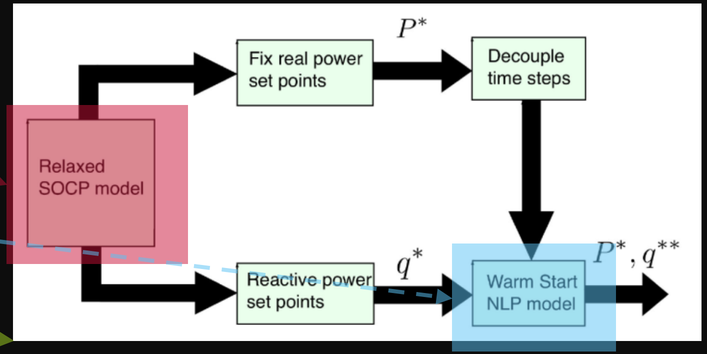
# How do others 'scalably' solve it? – Hierarchical Optimization, Approximated (Relaxed) Top Level Step

Specifically,

- The full optimization problem is divided into hierarchies, with the top hierarchy problem solving for an approximate (relaxed) model of the full horizon MINLP (NLP) problem.
- Once the top-level problem is solved and set-points for temporally-coupling decision variables are obtained, the same decision variables are kept as constant for every single time-step's OPF formulation.
- The now temporally-decoupled Single-Period Optimal Power Flow Problems are formulated as SOCP (NLP) in a parallel manner.
- *[nazir2018 (nazir2020)]*




Inner SOCP call within an outer MILP loop




Inner NLP call within an outer SOCP loop

# How do others 'scalably' solve it? – Some sort of Algorithms involving Primal Dual Decomposition of the Constraints containing Variables from different time-steps

Generator Ramping Constraints

$$\underline{P_{ramp}} \leq P_G^t - P_G^{t-\Delta t} \leq \overline{P_{ramp}}$$

$$B^t = B^{t-\Delta t} + \left(\eta_c P_c^t - \eta_d P_d^t\right) \Delta t$$

Battery SOC Constraint

Not fully appreciated (understood) by me yet. But here are two instances:

- In [agarwal2022] Their transmission system has many kinds of constraints, but only two temporally coupling constraints – Generator Ramping (one upper bound inequality and one lower bound inequality for each generator for each 'time-boundary') and Battery SOC (one equality for each battery for each time-boundary). All other equality/inequality constraints are constraints *within* a particular time-step, and thus are also called as *uncoupled constraints*.
- They then formulate a Lagrangian Function for the full OPF problem. After grouping 'like' terms together (as in whether they are temporally coupled or uncoupled), they ended up with this equation, which has (a summation of) temporally coupled terms (which come arise from the three kinds of temporal constraints) and temporally uncoupled terms (all contained in the first term)
- **Finally, from the full horizon problem, a *t*-th time-step problem function** is carved out, which, outside the variables for that time-step, **only contains certain primal and dual variables outside the time-step,** including:
  - $P_G^{t-\Delta t}, B^{t-\Delta t}, \mu_{P_{ramp}}^{t+\Delta t}$ (both lower and upper bounds), $\lambda_B^{t+\Delta t}$

$$\mathcal{L}_{\text{MPOPF}} = \sum_{t=1}^{t_n} \mathcal{L}_{ACOPF}^t + \sum_{t=2}^{t_n} \overline{\mu_{P_{ramp}}}^t \cdot \left(P_G^t - P_G^{t-\Delta t} - \overline{P_{ramp}}\right) - \underline{\mu_{P_{ramp}}}^t \cdot \left(P_G^t - P_G^{t-\Delta t} - \underline{P_{ramp}}\right) + \sum_{t=2}^{t_n} \lambda_B^t \left(B^t - \left(B^{t-\Delta t} + \left(\eta_c P_c - \eta_d P_d\right) \Delta \text{t}\right)\right) \quad (41)$$

Lagrangian Function representing the full MPOPF problem (all time-steps)

$$\begin{aligned} \mathcal{L}_{MPOPF}^t = {} & \mathcal{L}_{ACOPF}^t + \overline{\mu_{P_{ramp}}}^t \cdot \left(P_G^t - P_G^{t-\Delta t} - \overline{P_{ramp}}\right) \\ & - \underline{\mu_{P_{ramp}}}^t \cdot \left(P_G^t - P_G^{t-\Delta t} - \underline{P_{ramp}}\right) \\ & - P_G^t \left(\overline{\mu_{P_{ramp}}}^{t+\Delta t} - \underline{\mu_{P_{ramp}}}^{t+\Delta t}\right) \\ & + \lambda_B^t \left(B^t - \left(B^{t-\Delta t}\right.\right. \\ & \left.\left. + \left(\eta_c P_c - \eta_d P_d\right) \Delta t\right) - \lambda_B^{t+\Delta t} B^t \right. \end{aligned} \quad (47)$$

Lagrangian Function representing the OPF problem for the $t$-th time-step. Notice that the 'new' blocks are just terms from the 'existing' blocks but coming from the adjacent time-steps to $t$: $\{t - \Delta t, t + \Delta t\}$

# How do others 'scalably' solve it? – Some sort of Algorithms involving Primal Dual Decomposition of the Constraints containing Variables from different time-steps

- Finally, from the full horizon problem, a ***t*-th time-step problem function** is carved out, which, outside the variables for that time-step, **only contains certain primal and dual variables outside the time-step**, including:
  - Past Primal Variables: $P_{\mathrm{G}}^{t-\Delta t}, B^{t-\Delta t}$
  - Future Dual Variables: $\mu_{P_{ramp}}^{t+\Delta t}$ (both lower and upper bounds), $\lambda_B^{t+\Delta t}$
  - While the problem itself cannot be solved for without knowing both past and future variables, as part of their Differential Dynamic Programming (DDP) Paradigm, they do a Forwards Pass and a Backwards Pass, wherein in each pass, they assume one set of variables to be constant (initially guessed, or possibly known from a previous computation).

$$\begin{aligned}\mathcal{L}_{MPOPF}^{t} = \mathcal{L}_{ACOPF}^{t} &+ \overline{\mu_{P_{ramp}}}^{t} \cdot \left(P_G^t - P_G^{t-\Delta t} - \overline{P_{ramp}}\right. \\ &- \underline{\mu_{P_{ramp}}}^{t} \cdot \left(P_G^t - P_G^{t-\Delta t} - \underline{P_{ramp}}\right) \\ &- P_G^t \left(\overline{\mu_{P_{ramp}}}^{t+\Delta t} - \underline{\mu_{P_{ramp}}}^{t+\Delta t}\right) \\ &+ \lambda_B^t (B^t - (B^{t-\Delta t} \\ &+ (\eta_c P_c - \eta_d P_d)\,\Delta t) - \lambda_B^{t+\Delta t} B^t \qquad (47\end{aligned}$$

Lagrangian Function representing the OPF problem for the $t$-th time-step.

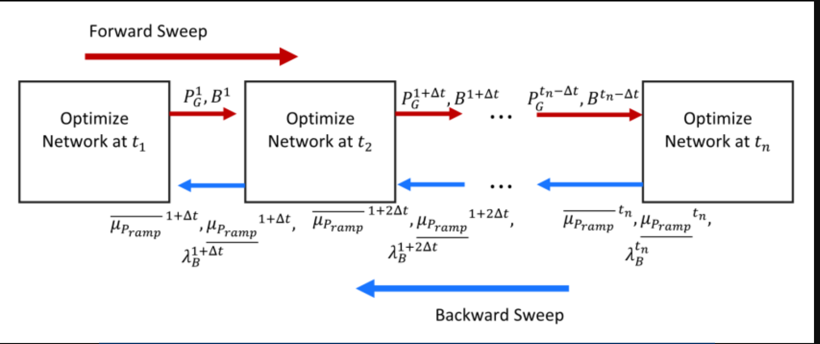
# How do others 'scalably' solve it? – Some sort of Algorithms involving Primal Dual Decomposition of the Constraints containing Variables from different time-steps

- Past Primal Variables: $P_G^{t-\Delta t}, B^{t-\Delta t}$
- Future Dual Variables: $\mu_{P_{ramp}}^{t+\Delta t}$ (both lower and upper bounds), $\lambda_B^{t+\Delta t}$

- Specifically, in Forward Pass, they use the solved-for values of the past primal variables i.e. from the $t-\Delta t$-th step as knowns for the problem at the $t$-th step, and assume the last-known/initially-guessed values of the dual variables from the future i.e. from the $t+\Delta t$-th step as constants, in order to fully solve the problem at the $t$-th step.
- Once the values of the primal variables values for the $t$-th step is known, they proceed one step *forward*, this time using those as the values of the past primal variables, again assuming the future dual variables as some constants.
- This forward pass process continues till the end-of-horizon (last time-step, let's call it $T$).

$$
\begin{aligned}
\mathcal{L}_{MPOPF}^{t} = \mathcal{L}_{ACOPF}^{t} &+ \overline{\mu_{P_{ramp}}}^{t} \cdot \left(P_G^t - P_G^{t-\Delta t} - \overline{P_{ramp}}\right. \\
&- \underline{\mu_{P_{ramp}}}^{t} \cdot \left(P_G^t - P_G^{t-\Delta t} - \underline{P_{ramp}}\right) \\
&- P_G^t \left(\overline{\mu_{P_{ramp}}}^{t+\Delta t} - \underline{\mu_{P_{ramp}}}^{t+\Delta t}\right) \\
&+ \lambda_B^t \left(B^t - \left(B^{t-\Delta t}\right.\right. \\
&+ \left.\left.\left(\eta_c P_c - \eta_d P_d\right) \Delta t\right)\right) - \lambda_B^{t+\Delta t} B^t \qquad (47
\end{aligned}
$$

Lagrangian Function representing the OPF problem for the $t$-th time-step.




Scheme for Forward Pass and Backward Pass (here called 'Sweep' with no difference)

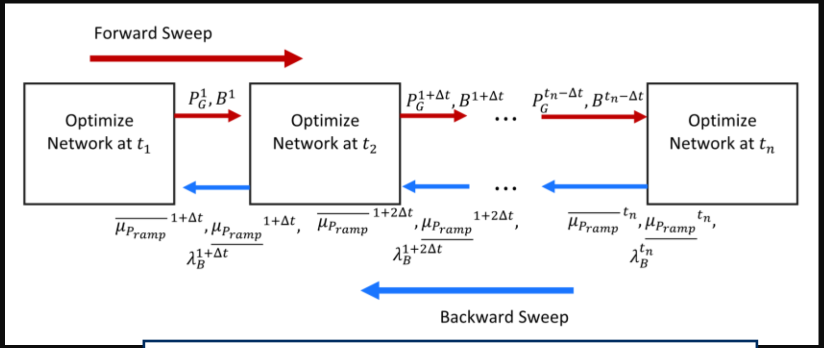
# How do others ‘scalably’ solve it? – Some sort of Algorithms involving Primal Dual Decomposition of the Constraints containing Variables from different time-steps

- Past Primal Variables: $P_G^{t-\Delta t}, B^{t-\Delta t}$
- Future Dual Variables: $\mu_{P_{ramp}}^{t+\Delta t}$ (both lower and upper bounds), $\lambda_B^{t+\Delta t}$

- Once a forward pass is completed, i.e. *the cost of the forward trajectory is constructed.* It is time to solve for the gradients that will update the dual variables for the next forward pass.
- Thus after computation of these gradients, the next set of dual variables $\mu_{P_{ramp}}^{t+\Delta t}, \lambda_B^{t+\Delta t}$ is computed.
- The process of Forward Pass and Backward Pass updates is carried on until *convergence* is achieved (updates no longer being significant).

$$
\begin{aligned}
\mathcal{L}^t_{MPOPF} = \mathcal{L}^t_{ACOPF} &+ \overline{\mu_{P_{ramp}}}^t \cdot \left(P_G^t - P_G^{t-\Delta t} - \overline{P_{ramp}}\right) \\
&- \underline{\mu_{P_{ramp}}}^t \cdot \left(P_G^t - P_G^{t-\Delta t} - \underline{P_{ramp}}\right) \\
&- P_G^t \left(\overline{\mu_{P_{ramp}}}^{t+\Delta t} - \underline{\mu_{P_{ramp}}}^{t+\Delta t}\right) \\
&+ \lambda_B^t \left(B^t - (B^{t-\Delta t} \right. \\
&\left. + (\eta_c P_c - \eta_d P_d)\,\Delta t\right) - \lambda_B^{t+\Delta t} B^t \qquad (47)
\end{aligned}
$$

Lagrangian Function representing the OPF problem for the $t$-th time-step.




Scheme for Forward Pass and Backward Pass (here called ‘Sweep’ with no difference)

$$
\begin{aligned}
\nabla_{P_G^t} \mathcal{L}_{MPOPF} &= \nabla_{P_G^t} \mathcal{L}^t_{APOPF} \\
&- \left(\overline{\mu_{P_{ramp}}}^{t+\Delta t} - \underline{\mu_{P_{ramp}}}^{t+\Delta t}\right) \qquad (49)
\end{aligned}
$$

$$
\nabla_{B^t} \mathcal{L}_{MPOPF} = \nabla_{B^t} \mathcal{L}^t_{ACOPF} - \lambda_B^{t+\Delta t} \qquad (50)
$$

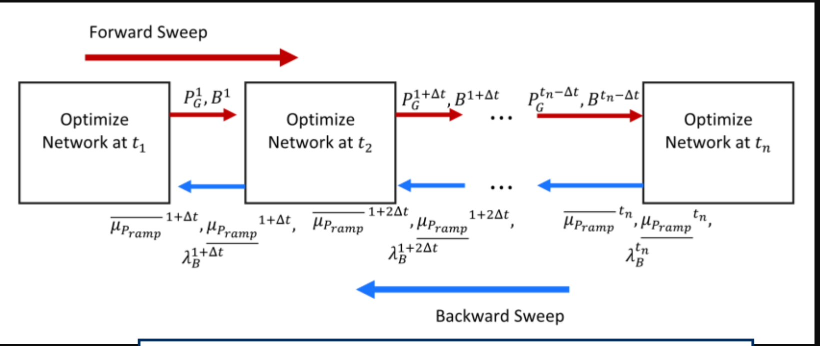
Equations used for updating the future dual variables as part of the Backward Pass

# How do others 'scalably' solve it? – Some sort of Algorithms involving Primal Dual Decomposition of the Constraints containing Variables from different time-steps

- Thus after computation of these gradients, the next set of dual variables $\overline{\mu_{P_{ramp}}}^{t+\Delta t}$, $\underline{\lambda_B^{t+\Delta t}}$ is computed.
- The process of Forward Pass and Backward Pass updates is carried on until *convergence* is achieved (updates no longer being significant).

$$
\begin{aligned}
\mathcal{L}^t_{MPOPF} = \mathcal{L}^t_{ACOPF} &+ \overline{\mu_{P_{ramp}}}^t \cdot \left(P^t_G - P^{t-\Delta t}_G - \overline{P_{ramp}}\right) \\
&- \underline{\mu_{P_{ramp}}}^t \cdot \left(P^t_G - P^{t-\Delta t}_G - \underline{P_{ramp}}\right) \\
&- P^t_G \left(\overline{\mu_{P_{ramp}}}^{t+\Delta t} - \underline{\mu_{P_{ramp}}}^{t+\Delta t}\right) \\
&+ \lambda^t_B (B^t - (B^{t-\Delta t} \\
&+ (\eta_c P_c - \eta_d P_d)\,\Delta t) - \lambda_B^{t+\Delta t} B^t
\end{aligned} \tag{47}
$$

Lagrangian Function representing the OPF problem for the $t$-th time-step.




Scheme for Forward Pass and Backward Pass (here called 'Sweep' with no difference)

Done: I've understood the overall structure of a DDP implementation. Specifically, I've understood what is known, what is solved for, and what is assumed constant within a single time-period's OPF. I understand the 'Forward Pass' component.

**Current:** I'm **still studying** the derivation required to understand the **'Backward-Pass'** (how to use the gradient of the Lagrangian to get the dual variable updates). Once that is understood, I'll have the understanding of one way to temporally decouple MPOPF for my use-case (ADS) as well, in a manner which can be implemented in code.

$$
\begin{aligned}
\nabla_{P^t_G} \mathcal{L}_{MPOPF} = \nabla_{P^t_G} \mathcal{L}^t_{APOPF} \\
- \left(\overline{\mu_{P_{ramp}}}^{t+\Delta t} - \underline{\mu_{P_{ramp}}}^{t+\Delta t}\right)
\end{aligned} \tag{49}
$$

$$
\nabla_{B^t} \mathcal{L}_{MPOPF} = \nabla_{B^t} \mathcal{L}^t_{ACOPF} - \lambda_B^{t+\Delta t} \tag{50}
$$

Equations used for updating the future dual variables as part of the Backward Pass

## How do others 'scalably' solve it? – Some sort of Algorithms involving Primal Dual Decomposition of the Constraints containing Variables from different time-steps

- A paper utilizing ADMM for temporal decomposition: [pinto2020]
- Deficiencies:
  - I don't understand how ADMM works, because I don't understand how dual variables work.
  - To have a good understanding of dual variables, I have been reading up on theory [wolf2011]. I've gained some confidence in it but am yet to understand certain other key concepts thoroughly.

The plan is to understand dual variables [wolf2011]., understand ADMM, understand ADMM usage in OPF [peng2015] , ultimately understand ADMM in MPOPF [pinto2020]

.

Latest Slides End Here.
Previous Slides Follow.