WASHINGTON STATE
UNIVERSITY

Preliminary Exam Presentation

Scalable Multi-Period Optimal Power Flow for Active Distribution Systems

Aryan Ritwajeet Jha

Pursuing PhD (ECE) Power Systems

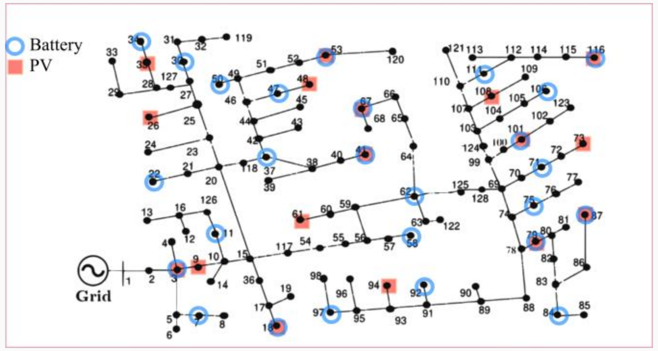
Introduction and Motivation

- What’s MPOPF?
- Real-world benefit?
- Why a whole PhD on it?
- Intended Contributions of this PhD

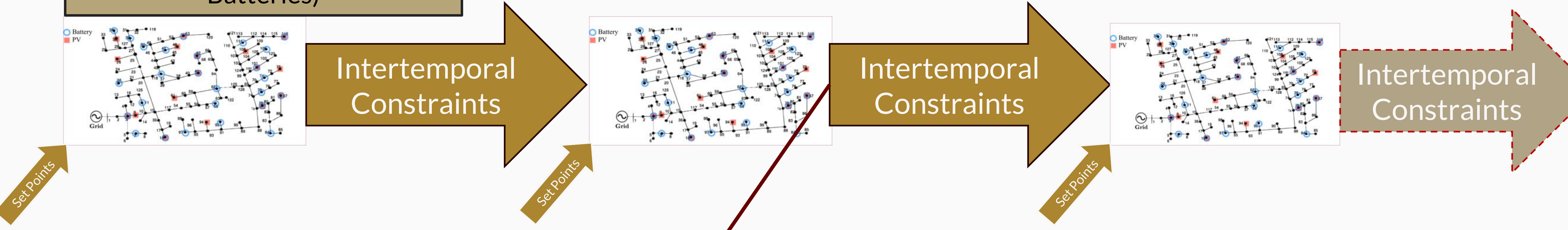
Introduction and Motivation

What's Multi-Period Optimal Power Flow (MPOPF)?

min. Desired Objective Function
subject to
Network Constraints
Engineering Constraints
Component Constraints (DERs, Batteries)



Controllable Components: Grid Edge Devices (GEDs) like Batteries and PVs spread throughout whose real and reactive dispatch may be set every time-period



Battery SOC Equation

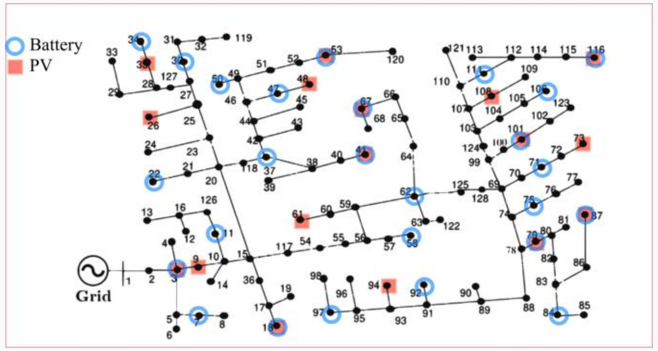
$$B_j^t = B_j^{t-1} + \Delta t \eta_c P_{c_j}^t - \Delta t \frac{1}{\eta_d} P_{d_j}^t$$

Due to these intertemporal constraints, the optimization problem size becomes T times larger, becoming more difficult to solve.

Introduction and Motivation

What's Multi-Period Optimal Power Flow (MPOPF)?

$min.$ Desired Objective Function
$subject\ to$
Network Constraints
Engineering Constraints
Component Constraints (DERs, Batteries)



Explain key assumptions – what DSO can see, observe, measure, control. What can they not control. Receding Horizon. What is an 'efficient' enough MPOPF?

Explain what is a temporal constraint

Battery SOC Equation

$$B_j^t = B_j^{t-1} + \Delta t \eta_c P_{c_j}^t - \Delta t \frac{1}{\eta_d} P_{d_j}^t$$

An example of

is

Introduction and Motivation

What Real-world benefit does research on MPOPF provide?

- The **MPOPF problem** is to be **solved routinely** (daily, hourly, 15-minutely) **by** Distribution System Operators (**DSOs**), Transmission System Operators (**TSOs**) to **schedule dispatch**
- While generally they may rely on approximated forms of the MPOPF problem, say Economic Dispatch or DC Power Flow, **increasing complexity and number of grid edge devices** can put
  1. obeyance of operational constraints at risk
  2. incur a lot of opportunity cost due to missed profits of operating at a suboptimal control point
- **Solving** the **true MP(AC-)OPF** problem – **if** possible **in** a **competitive time** would both be the **safest** and **most profitable** way to schedule dispatch for them

Introduction and Motivation

Is the MPOPF problem hard enough to warrant a PhD?

- Yes, the MP(AC-)OPF problem is inherently nonlinear and nonconvex – hard to solve for by optimization solvers unless the modeler can employ high quality approximation and/or relaxation techniques
- Still the MPOPF problem, as it expands in time-horizon T, can be very challenging to solve for due to its sheer size
- In such cases, **decomposition algorithms** may be employed exploit the weak coupling of individual time-step problems
- Putting out an algorithm isn't enough – engineers need strong convergence guarantees and competitive computational performance
- End-users need an easy-to-use framework to test the algorithm whose output in turn needs to be tested against trusted software

An example of

Introduction and Motivation

Intended Contributions of my PhD

- A framework for solving the MPOPF problem…
  - Which has a systematic procedure to model components of power distribution system in a manner faithful to their behaviour yet computationally efficient to solve for
  - That employs tailored algorithms which can exploit model's properties to come up with an even faster solution
  - Which has provision for comparison of output solution with those of trusted softwares, say OpenDSS
  - Whose procedure may be theoretically justified

An example of

PhD Execution Pipeline at a Glance

- Courses (Program of Study)
- Current Publications and intended future publication
- High level timeline

PhD Execution Pipeline at a Glance

Courses (Program of Study)

Course Number and Name	Semester	Instructor	Grade
E_E 507 Random Processes in Engineering	Fall 2022	Prof. Sandip Roy	A
E_E 521 Analysis of Power Systems	Fall 2022	Prof. Noel Schulz	A
E_E 523 Power Systems Stability	Spring 2023	Prof. Mani V. Venkatasubramanian	A
MATH 564 Convex and Nonlinear Optimization	Fall 2023	Prof. Tom Asaki	A
MATH 565 Nonsmooth Analysis and Optimization	Spring 2024	Prof. Tom Asaki	A-
CPT_S 530 Numerical Analysis[1]	Fall 2025	Prof. Alexander Panchenko	
E_E 582 Electrical Systems Modelling and Simulation[1]	Fall 2025	Prof. Seyedmilad Ebrahimi	
E_E 595 Directed Studies in Electrical Engineering[1]	Fall 2025	Prof. Rahul K. Gupta	

On track to be fulfilled by end of current semester (Fall 2025)

[1] currently taking this semester

PhD Execution Pipeline at a Glance

Current Publications and intended future publication

1. **Jha, A. R.**, Paul, S., & Dubey, A. . Spatially Distributed Multi-Period Optimal Power Flow with Battery Energy Storage Systems. 2024 56th North American Power Symposium (NAPS). IEEE. doi: 10.1109/NAPS61145.2024.10741846 [1]

2. **Jha, A. R.**, Paul, S., & Dubey, A. . Analyzing the Performance of Linear and Nonlinear Multi -Period Optimal Power Flow Models for Active Distribution Networks. 2025 IEEE North-East India International Energy Conversion Conference and Exhibition (NE-IECCE). IEEE. doi: 10.1109/NE-IECCE64154.2025.11183479 [8]

NAPS Paper

IAS Paper which can be extended to an IAS journal

- Very recently I've been able to solve the MPOPF problem using one of my decomposition techniques which I'm aiming to submit as extension to the IAS paper by December 2025.

Task List

1. Foundational Studies
2. **Modelling** Tradeoff Analysis
3. **Spatial Decomposition** Algorithm
4. **Temporal Decomposition** Algorithms
5. Extension to Three-Phase
6. Multiple-Source Optimal Power Flow (MS-OPF) Exploration

Modelling Tradeoff Analysis – Linear vs Nonlinear

Modelling Tradeoff Analysis – Linear vs Nonlinear

Key Formulation Differences

Table 1: BFM-NL vs LinDistFlow formulation comparison

Constraint	BFM-NL	LinDistFlow
Real power balance	$\sum_k P^t_{jk} - (P^t_{ij} - r_{ij}l^t_{ij}) = p^t_j$	$\sum_k P^t_{jk} - P^t_{ij} = p^t_j$
Reactive power balance	$\sum_k Q^t_{jk} - (Q^t_{ij} - x_{ij}l^t_{ij}) = q^t_j$	$\sum_k Q^t_{jk} - Q^t_{ij} = q^t_j$
Voltage drop (KVL)	$v^t_j = v^t_i - 2(r_{ij}P^t_{ij} + x_{ij}Q^t_{ij}) + (r^2_{ij} + x^2_{ij})l^t_{ij}$	$v^t_j = v^t_i - 2(r_{ij}P^t_{ij} + x_{ij}Q^t_{ij})$
Power-current relation	$(P^t_{ij})^2 + (Q^t_{ij})^2 = l^t_{ij}v^t_i$	Not modeled
Battery inverter	$(P^t_{B_j})^2 + (q^t_{B_j})^2 \leq S^2_{B_{R,j}}$	Hexagonal approx. [15]

Key Differences: BFM-NL includes loss terms $(r_{ij}l^t_{ij}, x_{ij}l^t_{ij})$ and quadratic constraints; LinDistFlow drops these for linearity, introducing approximation errors.

Common Constraints: Voltage limits $v^t_j \in [V^2_{min}, V^2_{max}]$, battery dynamics $B^t_j = B^{t-1}_j + \Delta t(\eta_c P^t_{c_j} - \eta_d^{-1} P^t_{d_j})$, power limits, SOC bounds.

- Add slides for BFM-NL vs LinDistFlow

Modelling Tradeoff Analysis – Linear vs Nonlinear

Key Results at a Glance

- **Optimality:** LinDistFlow introduces 0.4–0.7% gap for moderate-large systems
- **Reactive Power:** LinDistFlow systematically misallocates reactive resources
- **Feasibility:** Substation power errors up to 5% create operational risks
- **Speed:** 10–20× computational advantage enables rapid scenario analysis

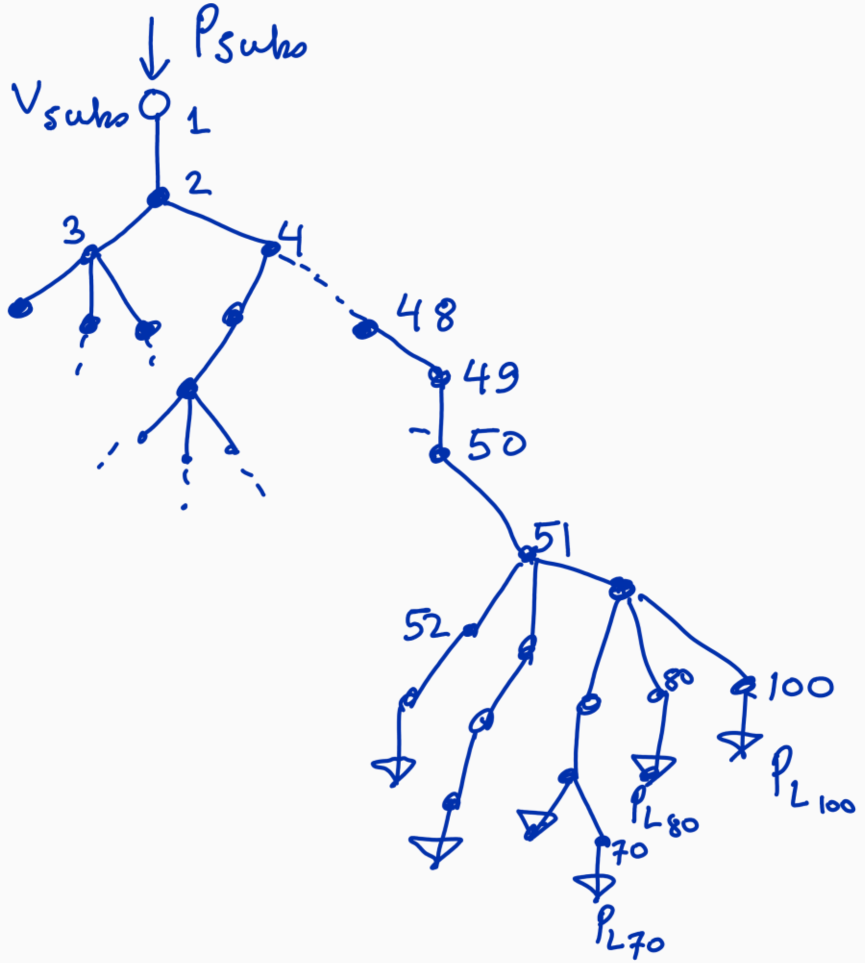
- Add slides for BFM-NL vs LinDistFlow

MPDOPF - Spatial Decomposition of the MPOPF problem using ENApp

- Intuition for Algorithm
- Results

MPDOPF - Spatially Decomposition MPOPF

Intuition for Spatial Decomposition

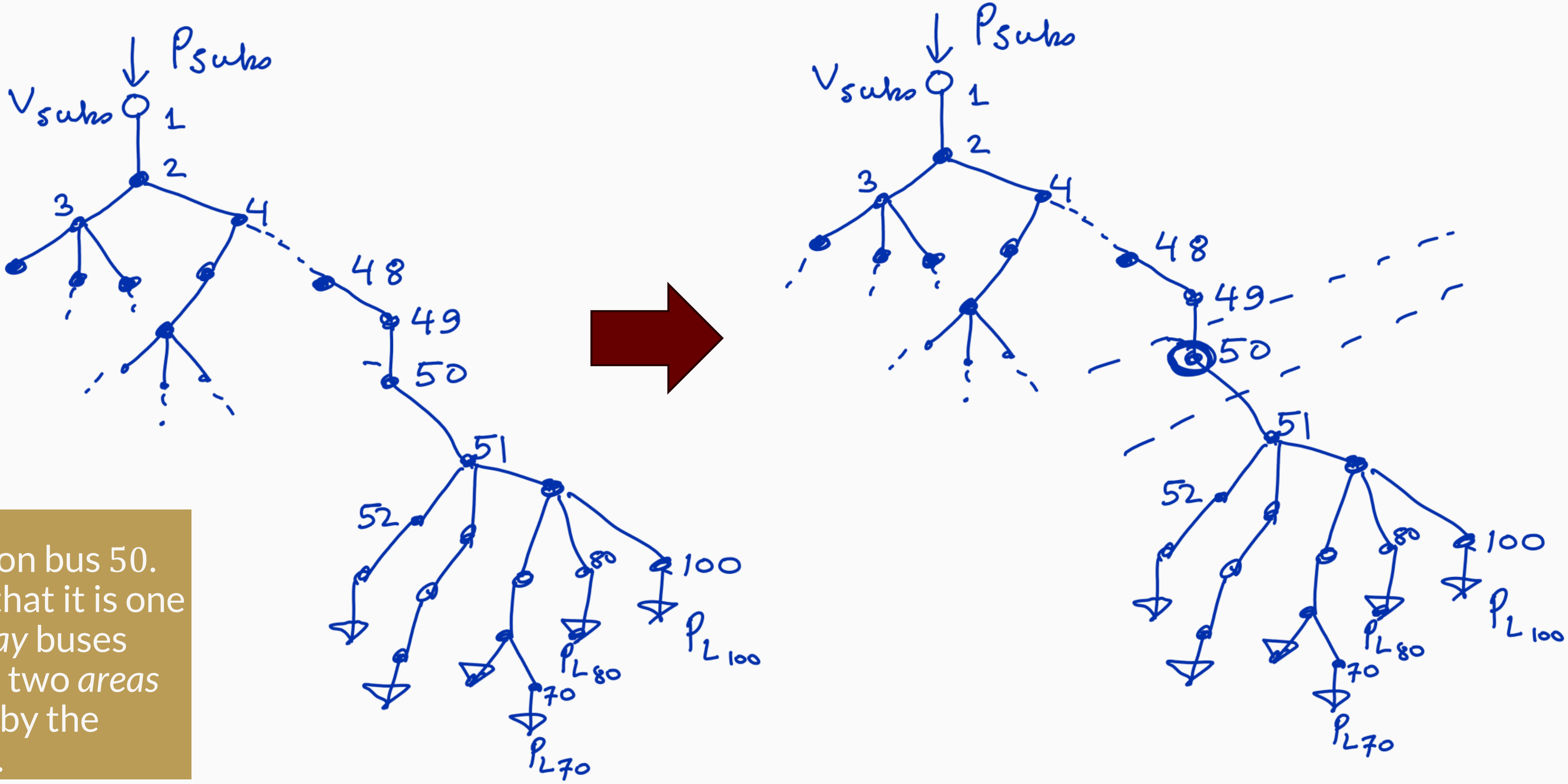


Imagine we have a 100 bus distribution system. Substation bus is 1. We wish to solve for its OPF, but would like to avoid solving the whole system in one go.

Let us focus on bus 50.

MPDOPF - Spatially Decomposition MPOPF

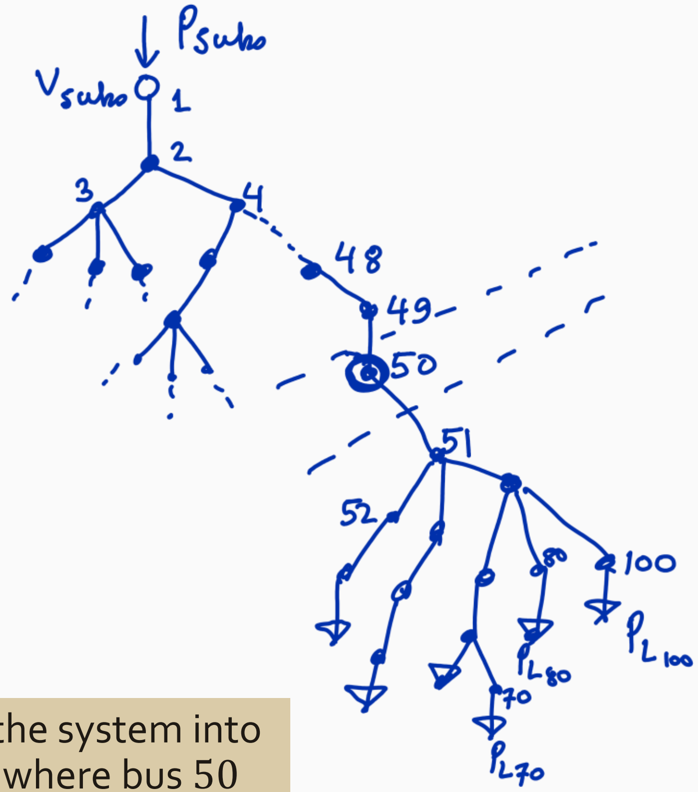
Intuition for Spatial Decomposition



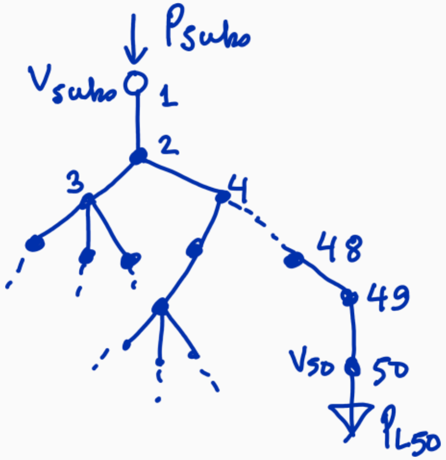
Let us focus on bus 50. We can see that it is one of the *gateway* buses between the two *areas* demarcated by the dashed lines.

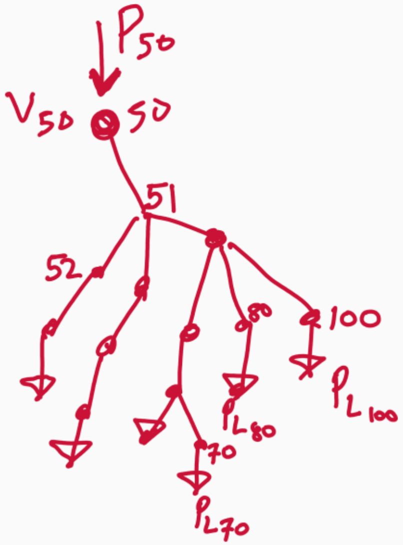
MPDOPF - Spatially Decomposition MPOPF

Intuition for Spatial Decomposition



We can divide the system into two *areas*, one where bus 50 behaves like a load node (whose load we don't know) and one where bus 50 behaves as a *substation* node (whose voltage we do don't know).

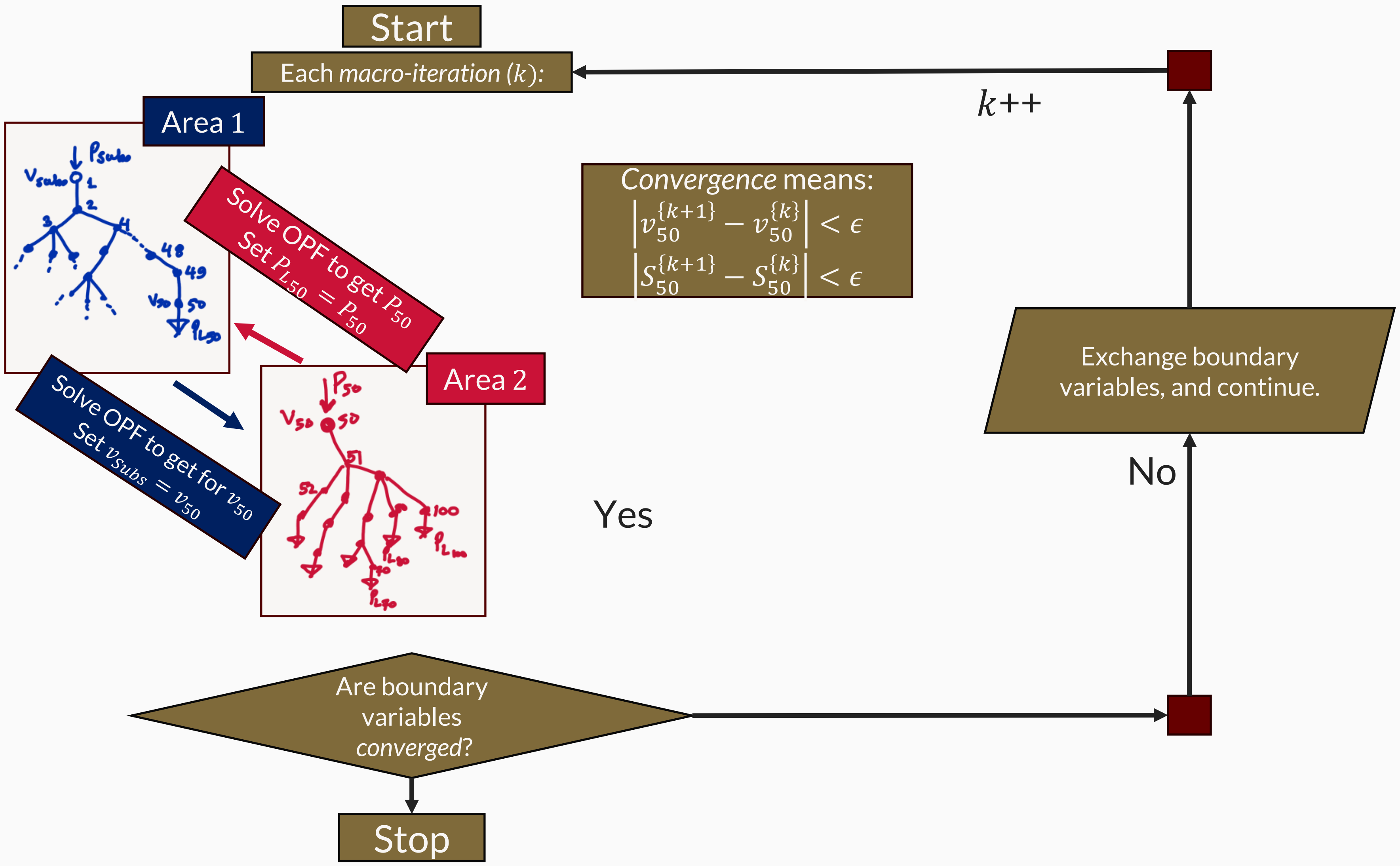




Now we have two OPF problems, each of half the size (and **less than half** the computation time) of the original problem

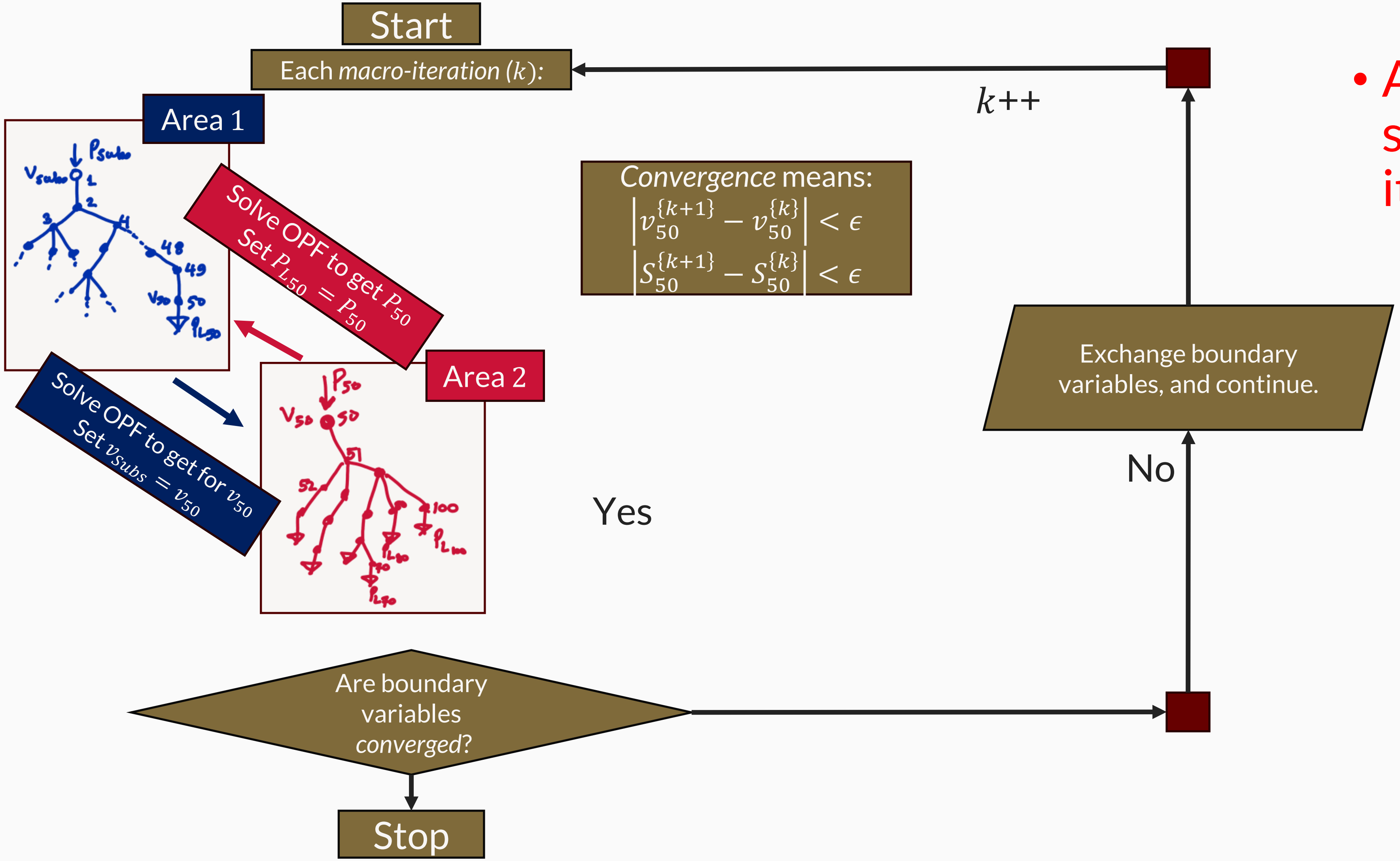
MPDOPF - Spatially Decomposition MPOPF

Intuition for Spatial Decomposition via OPF (single period)



MPDOPF - Spatially Decomposition MPOPF

Intuition for Spatial Decomposition via OPF (single period)



- Add final slide showing MPOPF iterations

MPDOPF - Spatially Decomposition MPOPF

Key Results at a Glance – Optimality and Performance for IEEE 123 1ph T=10

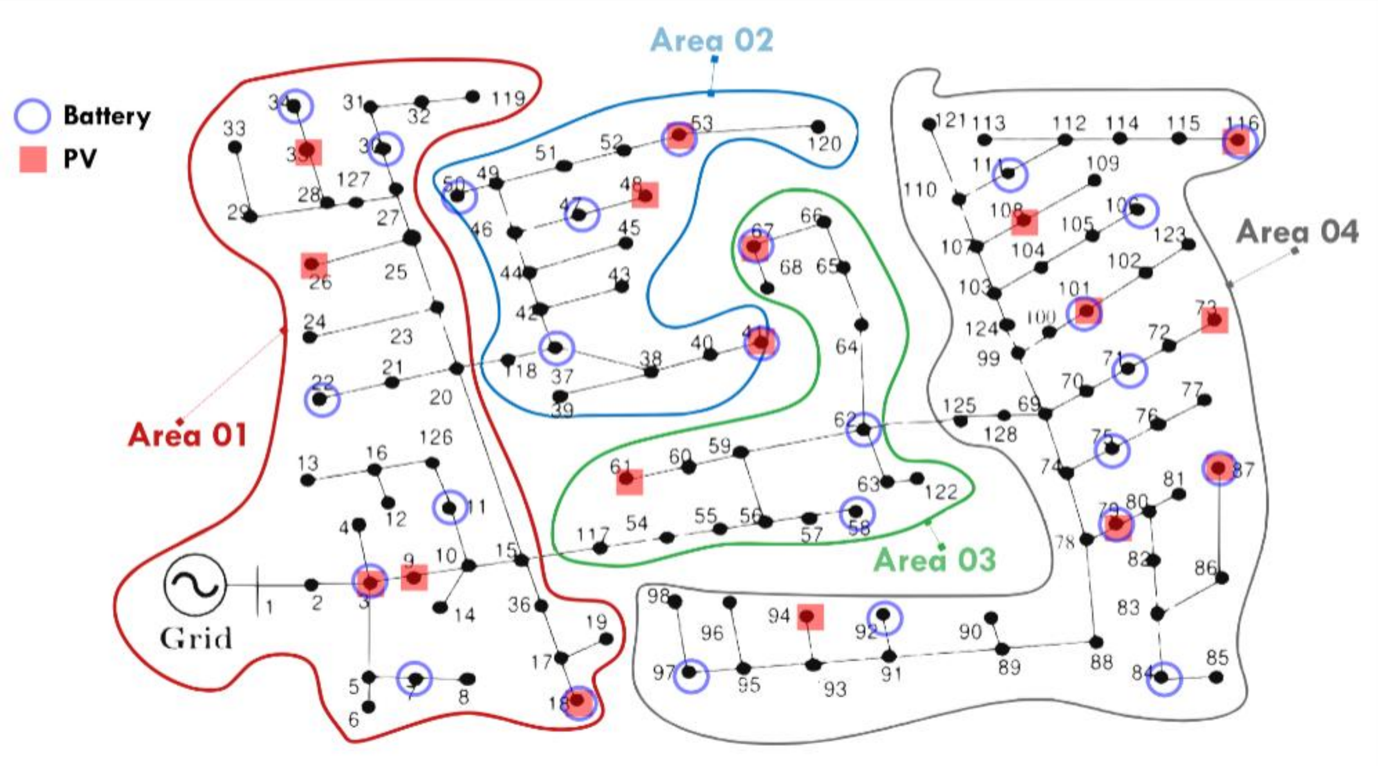
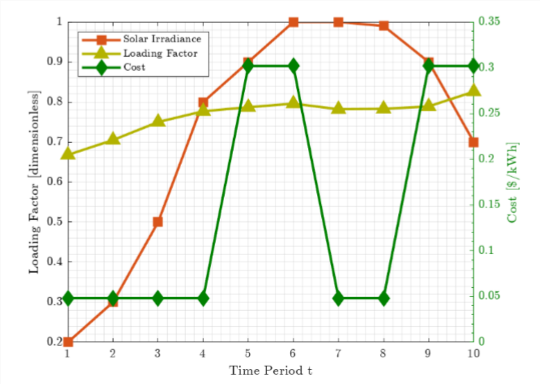


Figure 4: IEEE 123 node system divided into four areas

- Simulation Scenario Slides
- Optimization setup description

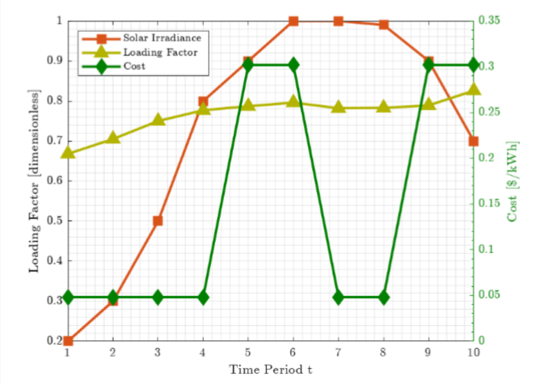


MPDOPF - Spatially Decomposition MPOPF

Key Results at a Glance – ACOPF feasibility analyses for IEEE 123 1ph T=10

Metric	MPDOPF	OpenDSS
Full horizon		
Substation real power (kW)	8544.04	8544.40
Line loss (kW)	148.94	148.87
Substation reactive power (kVAR)	1252.03	1243.36
Max. all-time discrepancy		
Voltage (pu)	0.0002	
Line loss (kW)	0.0132	
Substation power (kW)	0.4002	

- Simulation Scenario Slides
- Optimization setup description



DDP – Differential Dynamic Programming

- Results

DDP – Differential Dynamic Programming

Formulation for Forward Pass

$$\min \quad C^t P^t_{Subs} \tag{16}$$
$$+\alpha \sum_{j\in\mathscr{B}} \left\{(1-\eta_c)P^t_{c_j} + \left(\frac{1}{\eta_d}-1\right)P^t_{d_j}\right\}$$
$$+\sum_{j\in\mathscr{B}} \mu^{t+1}_{SOC_j}\{-B^t_j\}$$

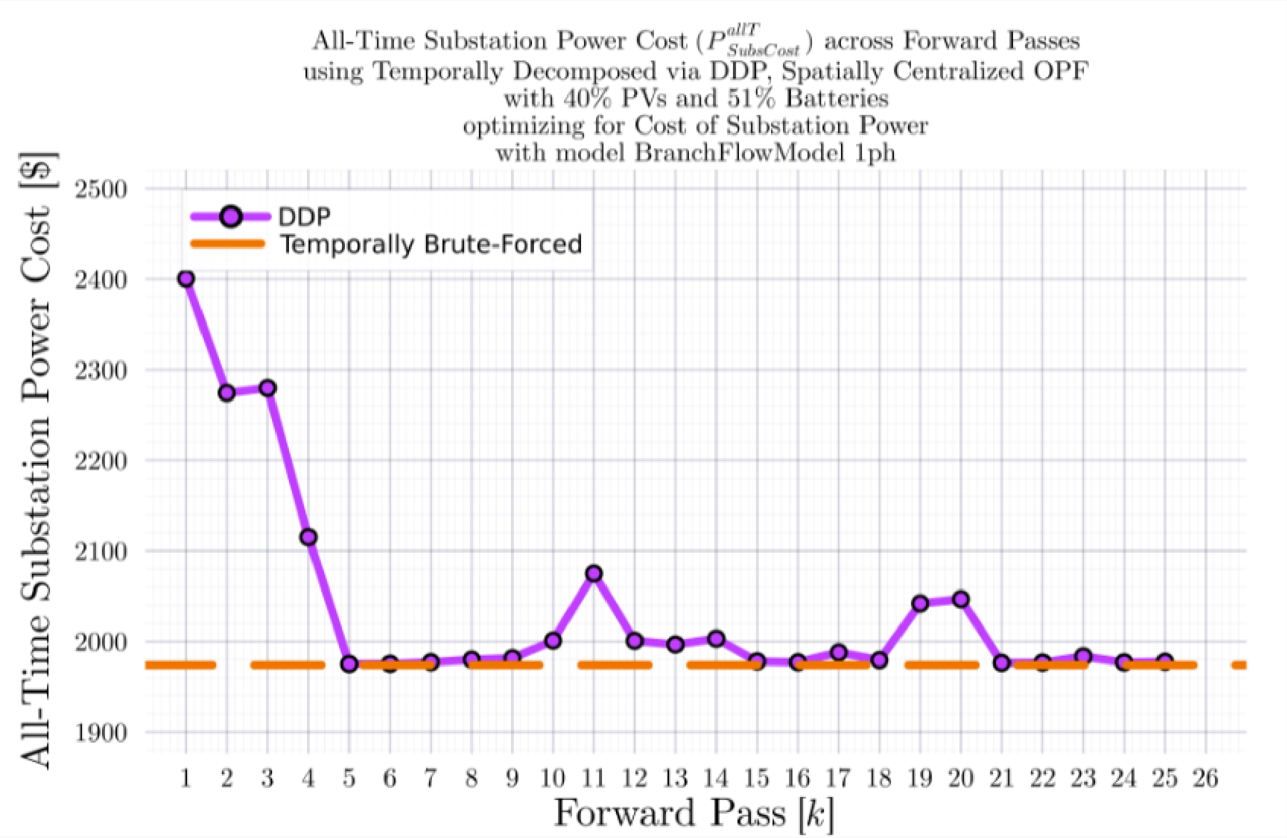
s.t.

$$0 = \sum_{(j,k)\in\mathscr{L}} \{P^t_{jk}\} - \{P^t_{ij} - r_{ij}l^t_{ij}\} - (P^t_{d_j} - P^t_{c_j}) - ({p_{D}}_j{}^t) + {p_L}_j{}^t \tag{17}$$
$$0 = \sum_{(j,k)\in\mathscr{L}} \{Q^t_{jk}\} - \{Q^t_{ij} - x_{ij}l^t_{ij}\} - (q^t_{D_j}) - (q^t_{B_j}) + {q_L}_j{}^t \tag{18}$$
$$0 = v^t_i - v^t_j - 2(r_{ij}P^t_{ij} + x_{ij}Q^t_{ij}) + \{r^2_{ij} + x^2_{ij}\}\, l^t_{ij} \tag{19}$$
$$0 = (P^t_{ij})^2 + (Q^t_{ij})^2 - v^t_i l^t_{ij} \tag{20}$$
$$0 = B^t_j - \left\{B^{t-1}_j + \Delta t\eta_c P^t_{c_j} - \Delta t\frac{1}{\eta_d}P^t_{d_j}\right\} \tag{21}$$
$$0 \geq {B_{min}}_j{}^t - {B_j}^t \tag{22}$$
$$0 \geq {B_j}^t - {B_{Max}}_j{}^t \tag{23}$$
$$0 \geq \text{All other inequality constraints} \tag{24}$$

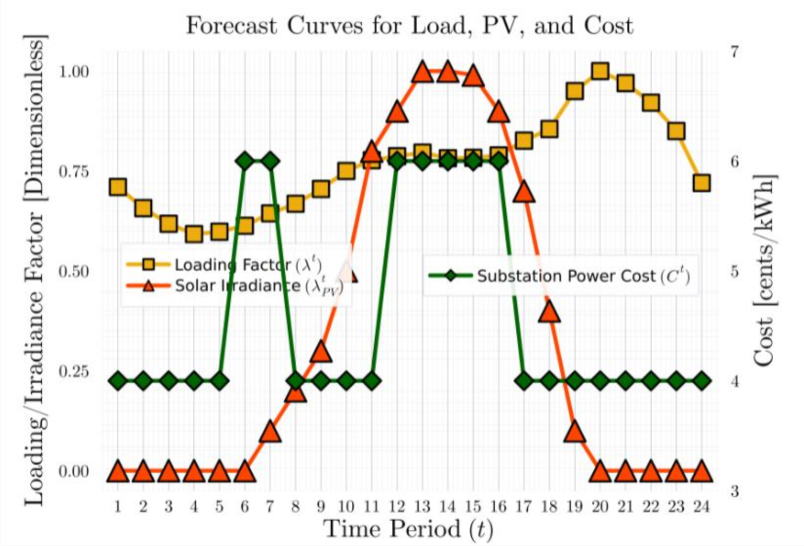
- Show derivation of this formulation!
- Show schematics for forward and backward pass exchange!

DDP – Differential Dynamic Programming

Key Results: IEEE123B_1ph BFM-NL with 40% PVs and 51% Batteries

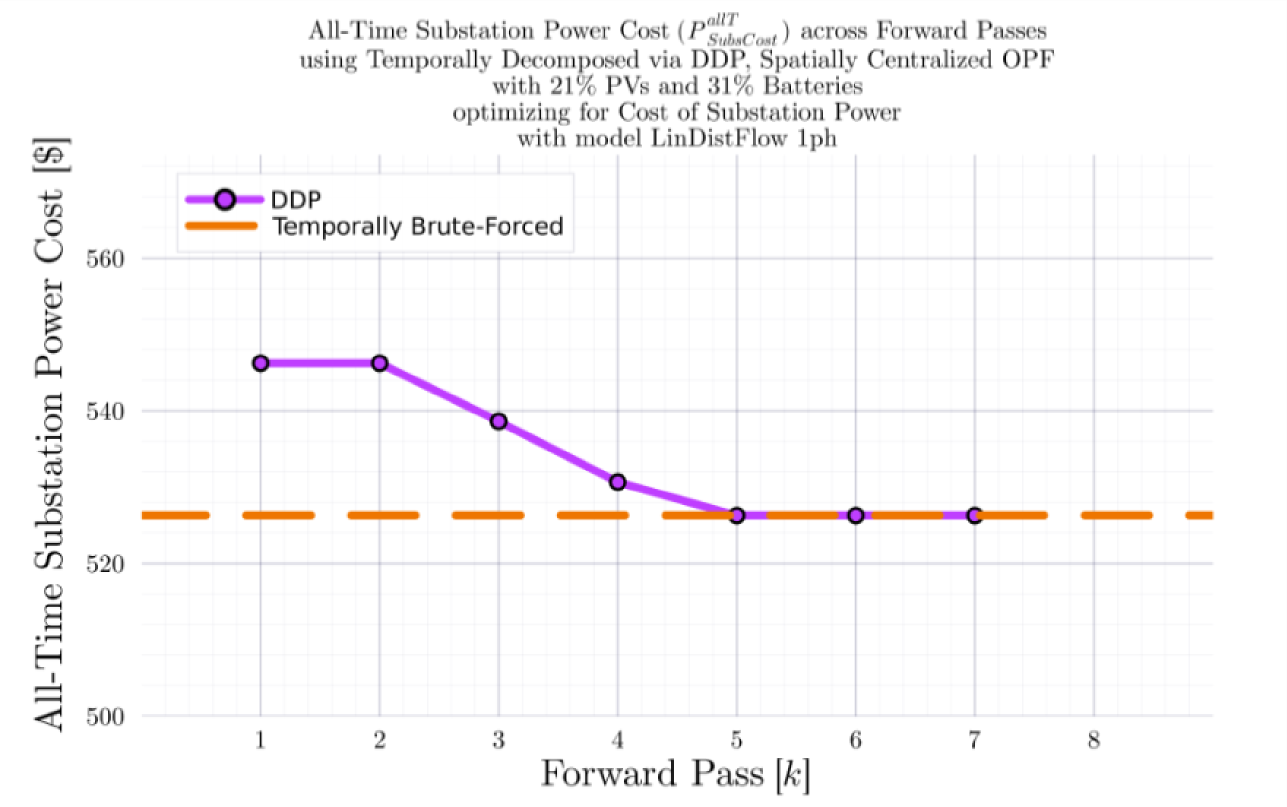


- Talk about current state of convergence

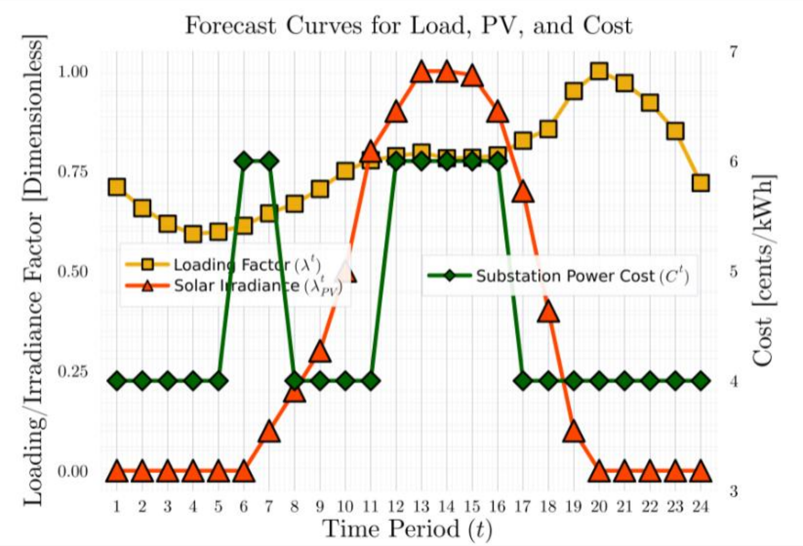


DDP – Differential Dynamic Programming

Key Results: IEEE730_1ph LDF with 21% PVs and 31% Batteries

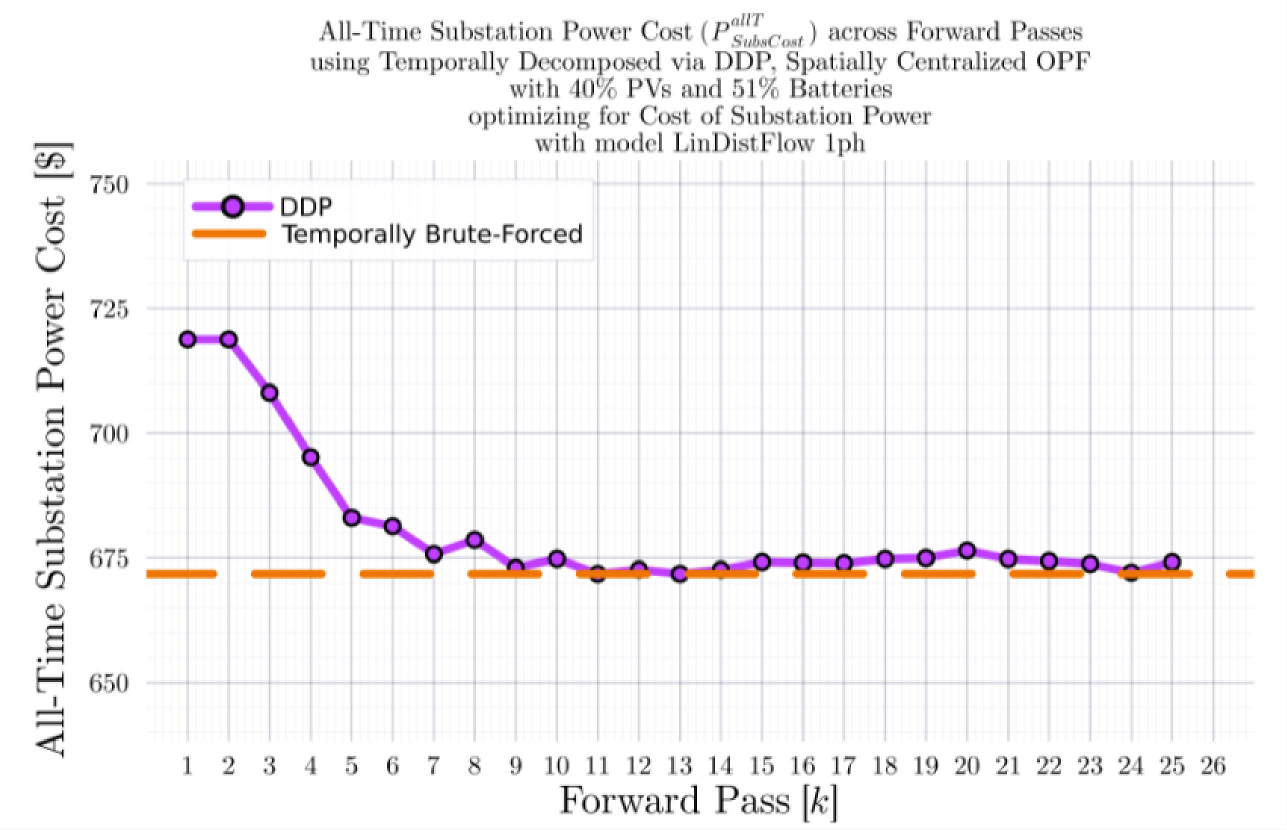


- Talk about current state of convergence

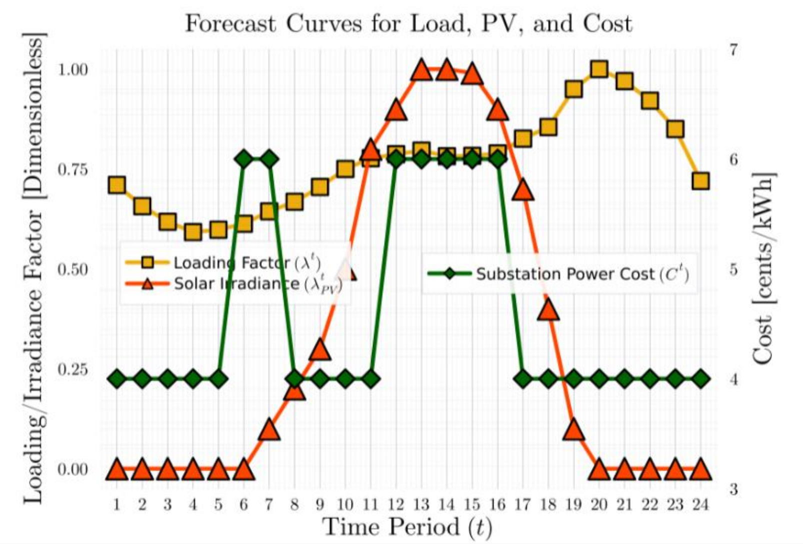


DDP – Differential Dynamic Programming

Key Results: IEEE123B_1ph LDF with 40% PVs and 51% Batteries



- Talk about current state of convergence
- All in all – currently still needs work



tADMM – temporal ADMM

- Formulation
- Results

tADMM – temporal ADMM

Algorithm formulation for copper plate example: Update 1 Primal Update

- Add a general diagram for how consensus admm works! References?
- Add a diagram for denoting tADMM workflow, showing what these variables mean!

$\mathbf{B}^{\mathbf{t_0}}$ (Blue): Local SOC variables for subproblem t_0

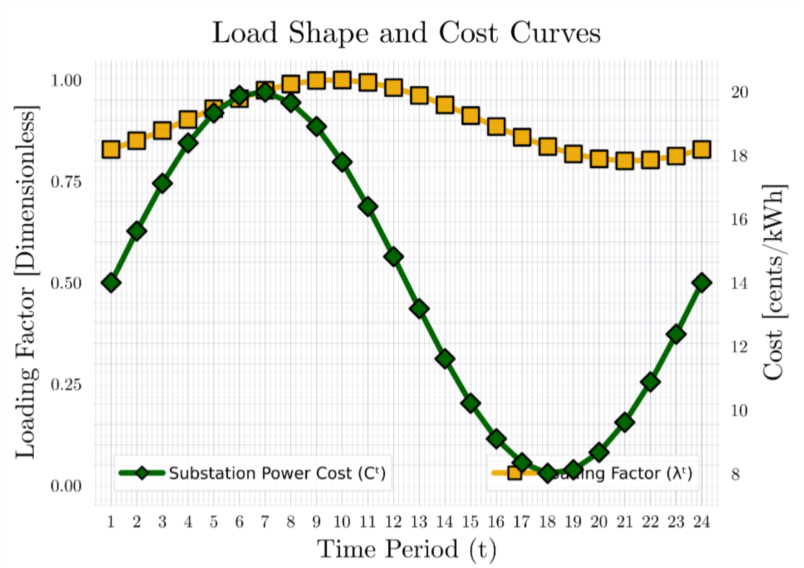
$\hat{\mathbf{B}}$ (Red): Global consensus SOC trajectory

$\mathbf{u}^{\mathbf{t_0}}$ (Green): Local scaled dual variables for subproblem t_0

tADMM – temporal ADMM

Algorithm formulation for copper plate example: Update 1 Primal Update

- Show a diagram for copper plate example
- Copper Plate = 1 Substation + 1 Battery supplying 1 Load for T time-steps



Input Curves:
$\lambda^t, C^t \; \forall t \in \mathrm{T}$

tADMM – temporal ADMM

Algorithm formulation for copper plate example: Update 1 Primal Update

For each subproblem $t_0 \in \{1, 2, \ldots, T\}$:

$$\min_{P_{\text{subs}}^{t_0}, P_B^{t_0}, \mathbf{B}^{t_0}} \quad C^{t_0} \cdot P_{\text{subs}}^{t_0} + C_B \cdot (P_B^{t_0})^2 + \frac{\rho}{2} \left\| \mathbf{B}^{t_0} - \hat{\mathbf{B}} + \mathbf{u}^{t_0} \right\|_2^2 \tag{50}$$

Subject to SOC Dynamics for Entire Trajectory:

$$\mathbf{B}^{t_0}[1] = B_0 - P_B^{t_0} \cdot \Delta t \tag{51}$$

$$\mathbf{B}^{t_0}[\mathbf{t}] = \mathbf{B}^{t_0}[\mathbf{t}-\mathbf{1}] - P_B^{t_0} \cdot \Delta t, \quad \forall t \in \{2, \ldots, T\} \tag{52}$$

$$P_{\text{subs}}^{t_0} + P_B^{t_0} = P_L[t_0] \tag{53}$$

$$-P_{B,R} \leq P_B^{t_0} \leq P_{B,R} \tag{54}$$

$$\underline{B} \leq \mathbf{B}^{t_0}[\mathbf{t}] \leq \overline{B}, \quad \forall t \in \{1, \ldots, T\} \tag{55}$$

- Primal Updates x T
- Each subproblem t_0 optimizes for its non-temporal constraints only
- However the temporal constraints (SOC trajectory) is computed for all time-steps
- This ensures that ADMM penalty term can compare the full trajectory $\boldsymbol{B}^{t0}$ with the consensus $\widehat{\boldsymbol{B}}$
- Uses last known values of $\widehat{\boldsymbol{B}}$ and $\boldsymbol{u}^{t0}$ and computes $\boldsymbol{B}^{t0}$

tADMM – temporal ADMM

Algorithm formulation for copper plate example: Update 2 Consensus Update

$$\hat{\mathrm{B}}[\mathrm{t}] = \operatorname{clamp}\left(\frac{1}{T}\sum_{t_0=1}^{T}\left(\mathrm{B}^{\mathrm{t_0}}[\mathrm{t}] + \mathrm{u}^{\mathrm{t_0}}[\mathrm{t}]\right), \underline{B}, \overline{B}\right) \quad \forall t \in \{1, 2, \ldots, T-1\} \tag{56}$$

- Consensus Update x 1
- Uses last known values of $\boldsymbol{B}^{t0}$ and $\boldsymbol{u}^{t0}$ and computes $\widehat{\boldsymbol{B}}$

tADMM – temporal ADMM

Algorithm formulation for copper plate example: Update 3 Dual Update

$$\mathbf{u}^{\mathbf{t_0}}[\mathrm{t}] := \mathbf{u}^{\mathbf{t_0}}[\mathrm{t}] + \left(\mathbf{B}^{\mathbf{t_0}}[\mathrm{t}] - \hat{\mathbf{B}}[\mathrm{t}]\right) \qquad (59)$$

$$\forall t_0 \in \{1, \ldots, T\}, \forall t \in \{1, \ldots, T\}$$

- Dual Update x T
- Uses last known values of $\boldsymbol{B}^{t0}$ and $\widehat{\boldsymbol{B}}$ and computes $\boldsymbol{u}^{t0}$

tADMM – temporal ADMM

Algorithm formulation for copper plate example: Convergence Criteria

$$\|r^k\|_2 = \frac{1}{|\mathscr{B}|}\sqrt{\sum_{j\in\mathscr{B}}\sum_{t=1}^{T}\left(\frac{1}{T}\sum_{t_0=1}^{T}\mathrm{B}_j^{t_0}[\mathrm{t}] - \hat{\mathrm{B}}_j[\mathrm{t}]\right)^2} \leq \epsilon_{\text{pri}} \tag{48}$$

1. Primal Residual
(Consensus Violation)

Each subproblem's own copy of SOC variables should resemble that of the global copy

$$\|s^k\|_2 = \frac{\rho}{|\mathscr{B}|}\sqrt{\sum_{j\in\mathscr{B}}\sum_{t=1}^{T}\left(\hat{\mathrm{B}}_j^{k}[\mathrm{t}] - \hat{\mathrm{B}}_j^{k-1}[\mathrm{t}]\right)^2} \leq \epsilon_{\text{dual}} \tag{49}$$

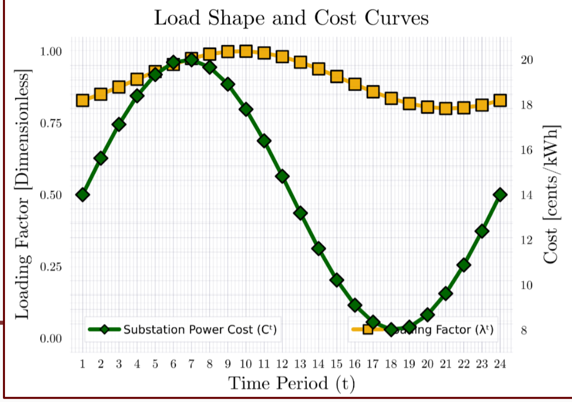
2. Dual Residual
(Consensus Change)

The consensus value should stabilize to a fixed point

Add: Explain how adaptive ADMM is implemented!

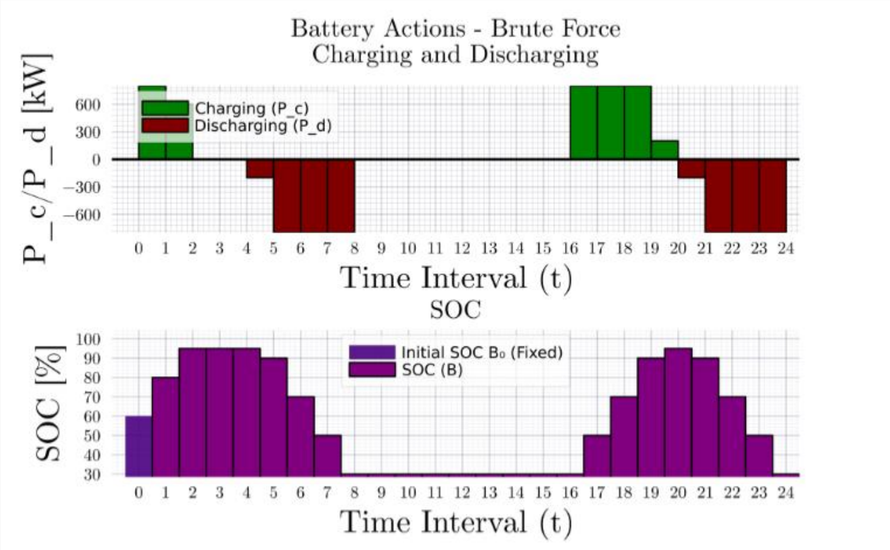
tADMM – temporal ADMM

MPOPF Simulation results for copper plate example

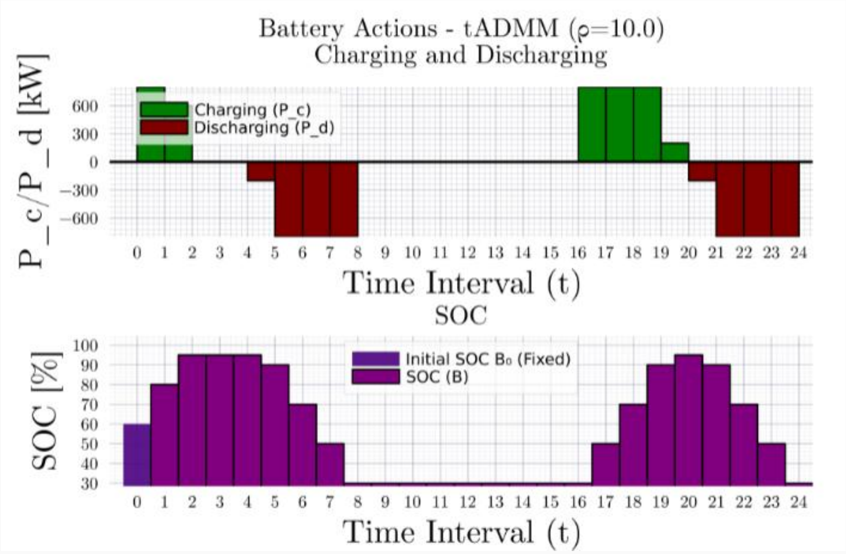


Input Curves

- **Battery Action plots** showcase that the results (control actions for battery dispatch) are identical



Brute Forced Battery Action Plots



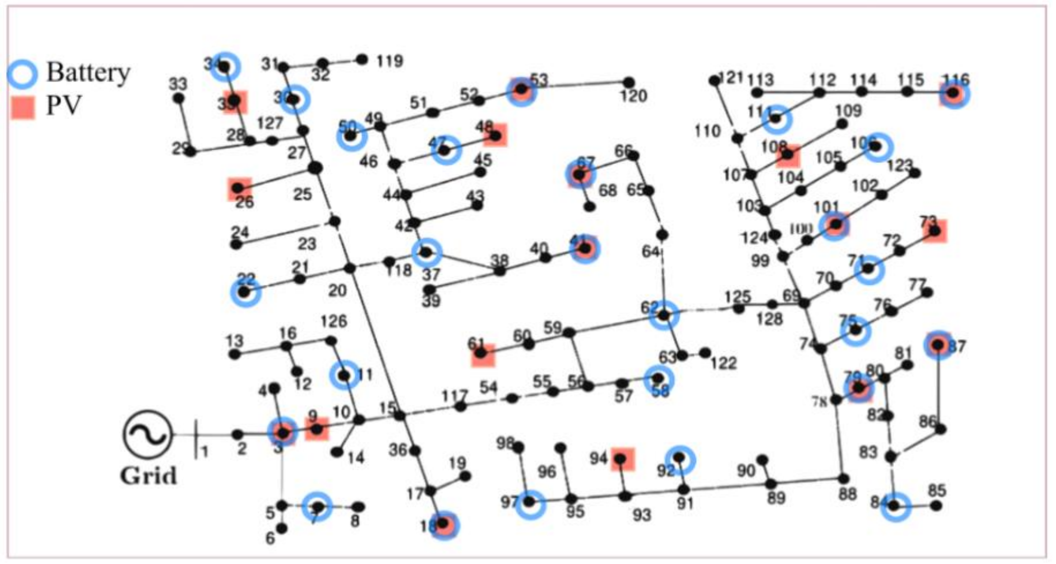
tADMM Battery Action Plots

All other results such as $P_{Subs}^t, P_{SubsCost}^t$ (not shown here) are also identical

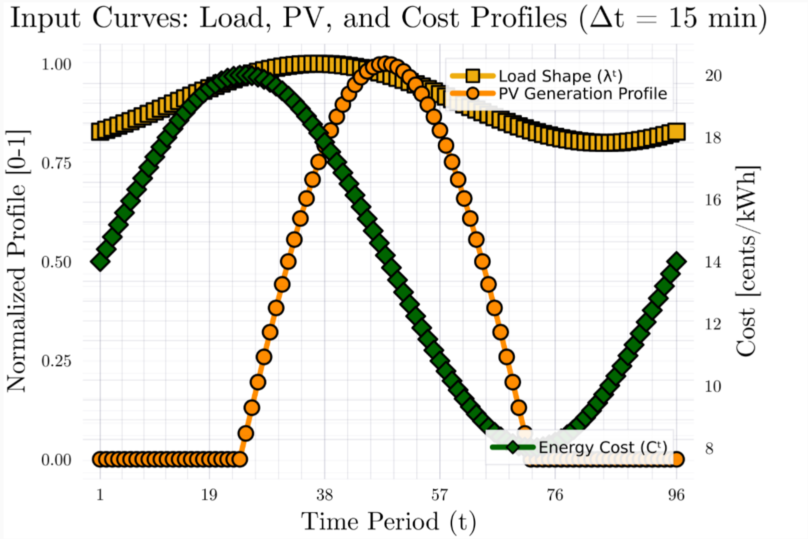
tADMM – temporal ADMM

Update modelling formulation for Full Network Model – Test Case Description

- **IEEE 123 1ph System** with **17 PVs** and **26 Batteries** – represented in both **LinDistFlow** (linear) **and BFM-NL** (nonlinear) models



IEEE123A_1ph Network



Input Curves: $\lambda_L^t, \lambda_{PV}^t, C^t \ \forall t \in 1\!:\!\mathrm{T}, \mathrm{T} = 96, \Delta t = 15min$

tADMM – temporal ADMM

Update modelling formulation for Full Network Model – Update 1 Primal Update

Objective Function: Cost of Borrowed Substation Power + Cost of Battery Operation + Augmented Lagrangian Term

$$\min_{\substack{P^{t_0}_{\text{Subs}}, Q^{t_0}_{\text{Subs}}, \\ P^{t_0}_{ij}, Q^{t_0}_{ij}, v^{t_0}_{j}, q^{t_0}_{D,j}, \\ P^{t}_{B,j}, \mathbf{B}^{\mathbf{t_0}}_{\mathbf{j}}[\mathbf{t}] \\ \forall j \in \mathscr{B}, t \in \mathscr{T}}} \quad c^{t_0} \cdot P^{t_0}_{\text{Subs}} \cdot P_{\text{BASE}} \cdot \Delta t + C_B \sum_{j \in \mathscr{B}} \left(P^{t_0}_{B,j}\right)^2 \cdot P^2_{\text{BASE}} \cdot \Delta t$$

$$+ \frac{\rho}{2} \sum_{j \in \mathscr{B}} \sum_{t=1}^{T} \left(\mathbf{B}^{\mathbf{t_0}}_{\mathbf{j}}[\mathbf{t}] - \hat{\mathbf{B}}_{\mathbf{j}}[\mathbf{t}] + \mathbf{u}^{\mathbf{t_0}}_{\mathbf{j}}[\mathbf{t}] \right)^2 \tag{34}$$

- Primal Updates x T
- This model is for LinDistFlow, but SOCP based BFM-NL has also been implemented by making the relevant adjustments (model not shown here)
- Uses last known values of $\widehat{\boldsymbol{B}_j}$ and $\boldsymbol{u}_j^{t0}$ and computes $\boldsymbol{B}_j^{t0}$ $\forall\, j \in B$

tADMM – temporal ADMM

Update modelling formulation for Full Network Model – Update 1 Primal Update

Objective Function: Cost of Borrowed Substation Power + Cost of Battery Operation + Augmented Lagrangian Term

$$\min_{\substack{P_{\text{Subs}}^{t_0}, Q_{\text{Subs}}^{t_0}, \\ P_{ij}^{t_0}, Q_{ij}^{t_0}, v_j^{t_0}, q_{D,j}^{t_0}, \\ P_{B,j}^{t}, \mathbf{B_j^{t_0}[t]} \\ \forall j \in \mathscr{B}, t \in \mathscr{T}}} c^{t_0} \cdot P_{\text{Subs}}^{t_0} \cdot P_{\text{BASE}} \cdot \Delta t + C_B \sum_{j \in \mathscr{B}} \left(P_{B,j}^{t_0}\right)^2 \cdot P_{\text{BASE}}^2 \cdot \Delta t$$

$$+ \frac{\rho}{2} \sum_{j \in \mathscr{B}} \sum_{t=1}^{T} \left(\mathbf{B_j^{t_0}[t]} - \mathbf{\hat{B}_j[t]} + \mathbf{u_j^{t_0}[t]}\right)^2 \tag{34}$$

t_0 specific Network Constraints: Temporally Uncoupled

$$\text{Real power balance (substation):} \quad P_{\text{Subs}}^{t_0} - \sum_{(1,j) \in \mathscr{L}_1} P_{1j}^{t_0} = 0 \tag{35}$$

$$\text{Real power balance (nodes):} \quad P_{ij}^{t_0} - \sum_{(j,k) \in \mathscr{L}} P_{jk}^{t_0} = P_{B,j}^{t_0} + p_{D,j}^{t_0} - p_{L,j}^{t_0}, \quad \forall (i,j) \in \mathscr{L}, \tag{36}$$

$$\text{Reactive power balance (substation):} \quad Q_{\text{Subs}}^{t_0} - \sum_{(1,j) \in \mathscr{L}_1} Q_{1j}^{t_0} = 0 \tag{37}$$

$$\text{Reactive power balance (nodes):} \quad Q_{ij}^{t_0} - \sum_{(j,k) \in \mathscr{L}} Q_{jk}^{t_0} = q_{D,j}^{t_0} - q_{L,j}^{t_0}, \quad \forall (i,j) \in \mathscr{L}, \tag{38}$$

$$\text{KVL constraints:} \quad v_i^{t_0} - v_j^{t_0} = 2(r_{ij} P_{ij}^{t_0} + x_{ij} Q_{ij}^{t_0}), \quad \forall (i,j) \in \mathscr{L} \tag{39}$$

$$\text{Voltage limits:} \quad (V_{\min,j})^2 \le v_j^{t_0} \le (V_{\max,j})^2, \quad \forall j \in \mathscr{N} \tag{40}$$

$$\text{PV reactive limits:} \quad -\sqrt{(S_{D,j})^2 - (p_{D,j}^{t_0})^2} \le q_{D,j}^{t_0} \le \sqrt{(S_{D,j})^2 - (p_{D,j}^{t_0})^2}, \quad \forall j \in \mathscr{D} \tag{41}$$

- Primal Updates x T
- This model is for LinDistFlow, but SOCP based BFM-NL has also been implemented by making the relevant adjustments (model not shown here)
- Uses last known values of $\widehat{B_j}$ and u_j^{t0} and computes B_j^{t0} $\forall\, j \in B$

tADMM – temporal ADMM

Update modelling formulation for Full Network Model – Update 1 Primal Update

Objective Function: Cost of Borrowed Substation Power + Cost of Battery Operation + Augmented Lagrangian Term

$$\min_{\substack{P^{t_0}_{\text{Subs}}, Q^{t_0}_{\text{Subs}},\\ P^{t_0}_{ij}, Q^{t_0}_{ij}, v^{t_0}_j, q^{t_0}_{D,j},\\ P^t_{B,j}, \mathbf{B}^{\mathbf{t_0}}_{\mathbf{j}}[\mathbf{t}]\\ \forall j\in\mathscr{B}, t\in\mathscr{T}}} c^{t_0}\cdot P^{t_0}_{\text{Subs}}\cdot P_{\text{BASE}}\cdot\Delta t + C_B\sum_{j\in\mathscr{B}}\left(P^{t_0}_{B,j}\right)^2\cdot P^2_{\text{BASE}}\cdot\Delta t + \frac{\rho}{2}\sum_{j\in\mathscr{B}}\sum_{t=1}^{T}\left(\mathbf{B}^{\mathbf{t_0}}_{\mathbf{j}}[\mathbf{t}]-\hat{\mathbf{B}}_{\mathbf{j}}[\mathbf{t}]+\mathbf{u}^{\mathbf{t_0}}_{\mathbf{j}}[\mathbf{t}]\right)^2 \quad (34)$$

t_0 specific Network Constraints: Temporally Uncoupled

$$\text{Real power balance (substation):}\quad P^{t_0}_{\text{Subs}} - \sum_{(1,j)\in\mathscr{L}_1} P^{t_0}_{1j} = 0 \quad (35)$$

$$\text{Real power balance (nodes):}\quad P^{t_0}_{ij} - \sum_{(j,k)\in\mathscr{L}} P^{t_0}_{jk} = P^{t_0}_{B,j} + p^{t_0}_{D,j} - p^{t_0}_{L,j}, \quad \forall (i,j)\in\mathscr{L}, \quad (36)$$

$$\text{Reactive power balance (substation):}\quad Q^{t_0}_{\text{Subs}} - \sum_{(1,j)\in\mathscr{L}_1} Q^{t_0}_{1j} = 0 \quad (37)$$

$$\text{Reactive power balance (nodes):}\quad Q^{t_0}_{ij} - \sum_{(j,k)\in\mathscr{L}} Q^{t_0}_{jk} = q^{t_0}_{D,j} - q^{t_0}_{L,j}, \quad \forall (i,j)\in\mathscr{L}, \quad (38)$$

$$\text{KVL constraints:}\quad v^{t_0}_i - v^{t_0}_j = 2(r_{ij}P^{t_0}_{ij} + x_{ij}Q^{t_0}_{ij}), \quad \forall (i,j)\in\mathscr{L} \quad (39)$$

$$\text{Voltage limits:}\quad (V_{\min,j})^2 \le v^{t_0}_j \le (V_{\max,j})^2, \quad \forall j\in\mathscr{N} \quad (40)$$

$$\text{PV reactive limits:}\quad -\sqrt{(S_{D,j})^2-(p^{t_0}_{D,j})^2} \le q^{t_0}_{D,j} \le \sqrt{(S_{D,j})^2-(p^{t_0}_{D,j})^2}, \quad \forall j\in\mathscr{D} \quad (41)$$

- Primal Updates x T
- This model is for LinDistFlow, but SOCP based BFM-NL has also been implemented by making the relevant adjustments (model not shown here)
- Uses last known values of $\widehat{B}_j$ and u_j^{t0} and computes $B_j^{t0}\ \forall\, j \in B$

Shared Temporally Coupled Constraints

$$\text{Initial SOC:}\quad \mathbf{B}^{\mathbf{t_0}}_{\mathbf{j}}[\mathbf{1}] = B_{0,j} - P^1_{B,j}\cdot\Delta t, \quad \forall j\in\mathscr{B} \quad (42)$$

$$\text{SOC trajectory:}\quad \mathbf{B}^{\mathbf{t_0}}_{\mathbf{j}}[\mathbf{t}] = \mathbf{B}^{\mathbf{t_0}}_{\mathbf{j}}[\mathbf{t-1}] - P^t_{B,j}\cdot\Delta t, \quad \forall t\in\{2,\dots,T\},\ j\in\mathscr{B} \quad (43)$$

$$\text{SOC limits:}\quad \text{SOC}_{\min,j}\cdot B_{\text{rated},j} \le \mathbf{B}^{\mathbf{t_0}}_{\mathbf{j}}[\mathbf{t}] \le \text{SOC}_{\max,j}\cdot B_{\text{rated},j}, \quad \forall t\in\mathscr{T},\ j\in\mathscr{B} \quad (44)$$

$$\text{Power limits:}\quad -P_{B,\text{rated},j} \le P^t_{B,j} \le P_{B,\text{rated},j}, \quad \forall t\in\mathscr{T},\ j\in\mathscr{B} \quad (45)$$

tADMM – temporal ADMM

Update modelling formulation for Full Network Model – Updates 2 and 3

$$\hat{\mathrm{B}}_{\mathrm{j}}[\mathrm{t}] = \mathrm{clamp}\left(\frac{1}{T}\sum_{t_0=1}^{T}\left(\mathrm{B}_{\mathrm{j}}^{\mathrm{to}}[\mathrm{t}] + \mathrm{u}_{\mathrm{j}}^{\mathrm{to}}[\mathrm{t}]\right), \underline{B}_j, \overline{B}_j\right) \tag{46}$$

- Consensus Updates x n_B
- Uses last known values of B_j^{t0} and u_j^{t0} and computes $\widehat{B_j}\ \forall\, j \in B$

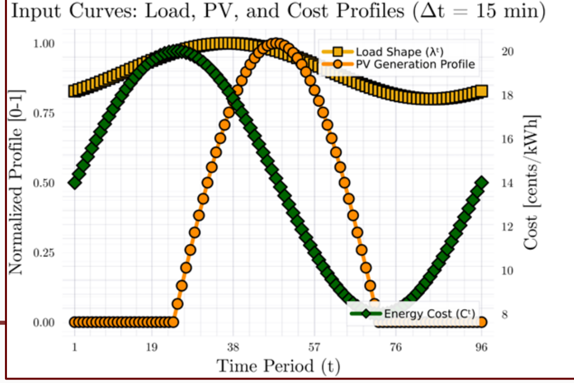
$$\mathrm{u}_{\mathrm{j}}^{\mathrm{to}}[\mathrm{t}] := \mathrm{u}_{\mathrm{j}}^{\mathrm{to}}[\mathrm{t}] + \left(\mathrm{B}_{\mathrm{j}}^{\mathrm{to}}[\mathrm{t}] - \hat{\mathrm{B}}_{\mathrm{j}}[\mathrm{t}]\right) \tag{47}$$

- Dual Updates x n_B
- Uses last known values of B_j^{t0} and $\widehat{B_j}$ and computes $u_j^{t0}\ \forall\, j \in B$

Both updates are pretty much the same as that for copper plate example, only differing in having to account for multiple batteries

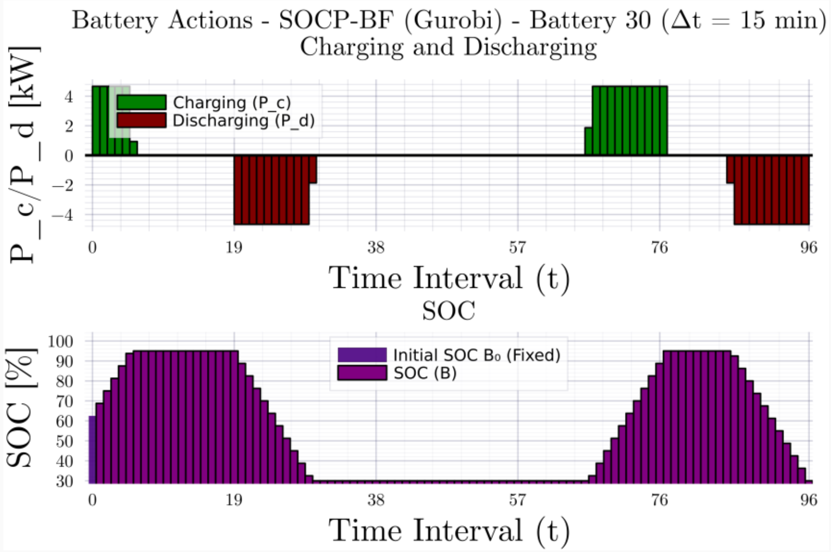
tADMM – temporal ADMM

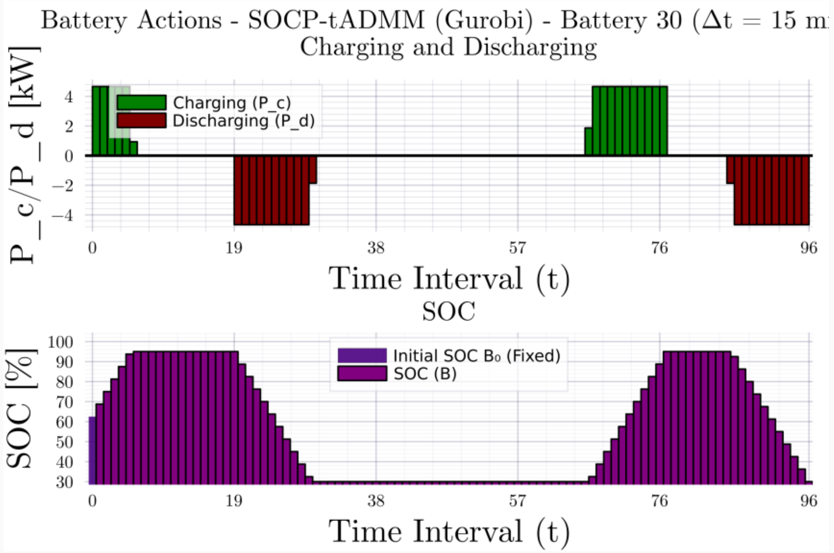
MPOPF Simulation results for IEEE123A_1ph – Battery Actions



Input Curves

- These results are for nonlinear model (SOCP relaxed BFM-NL)
- **Battery Action plots** showcase that the results (control actions for battery dispatch) are identical



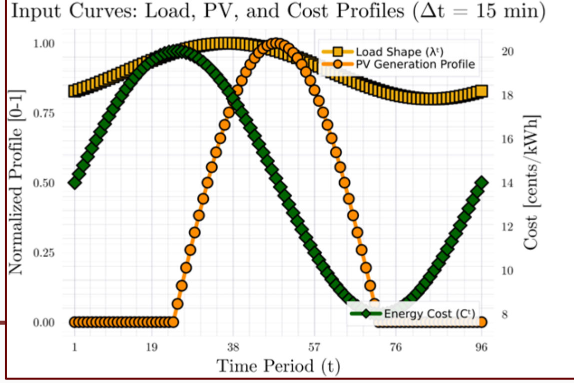


Brute Forced Battery Action Plots

tADMM Battery Action Plots

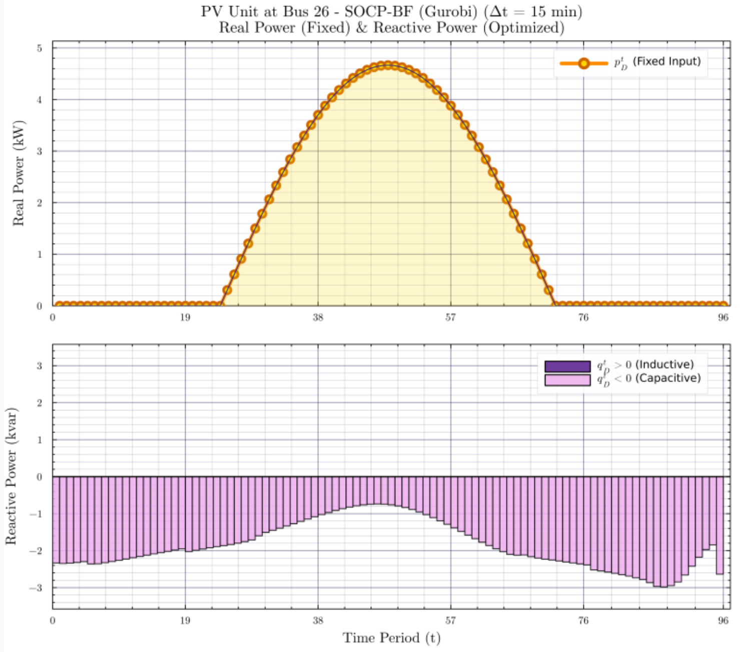
tADMM – temporal ADMM

MPOPF Simulation results for IEEE123A_1ph – PV Actions

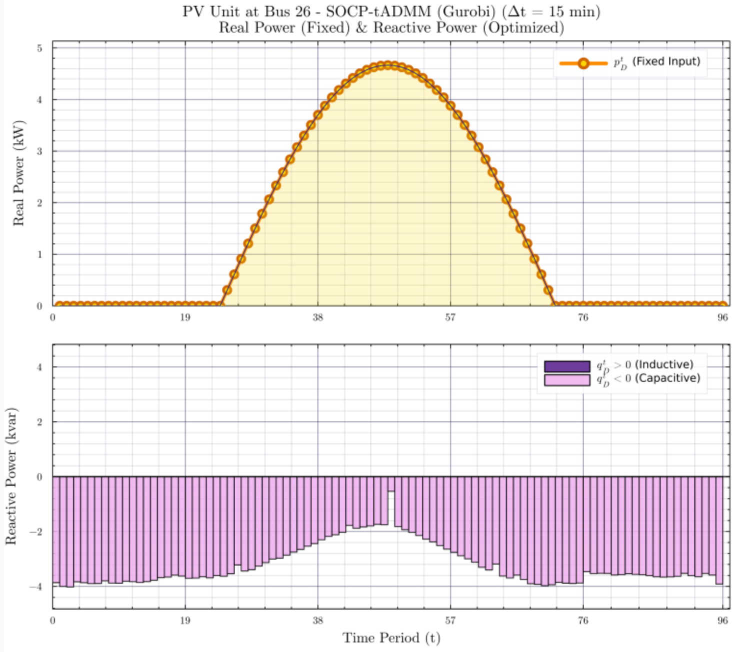


Input Curves

- **PV Reactive Power Dispatch plots** showcase are generally NOT necessarily identical



Note that PV real power dispatch is a fixed variable dependent only on λ^t_{PV} (fixed variable)

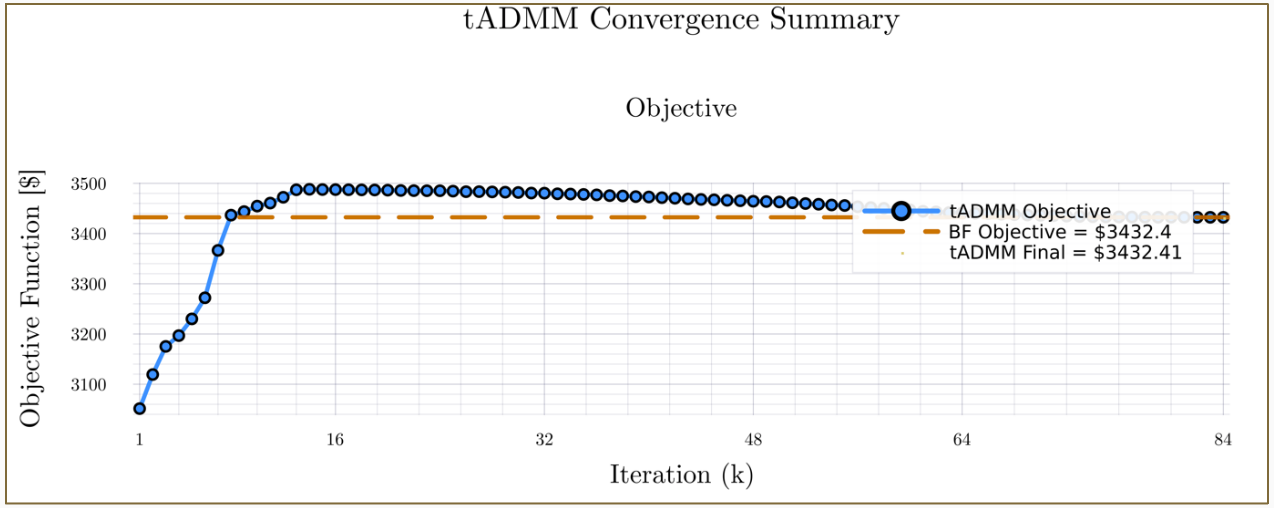


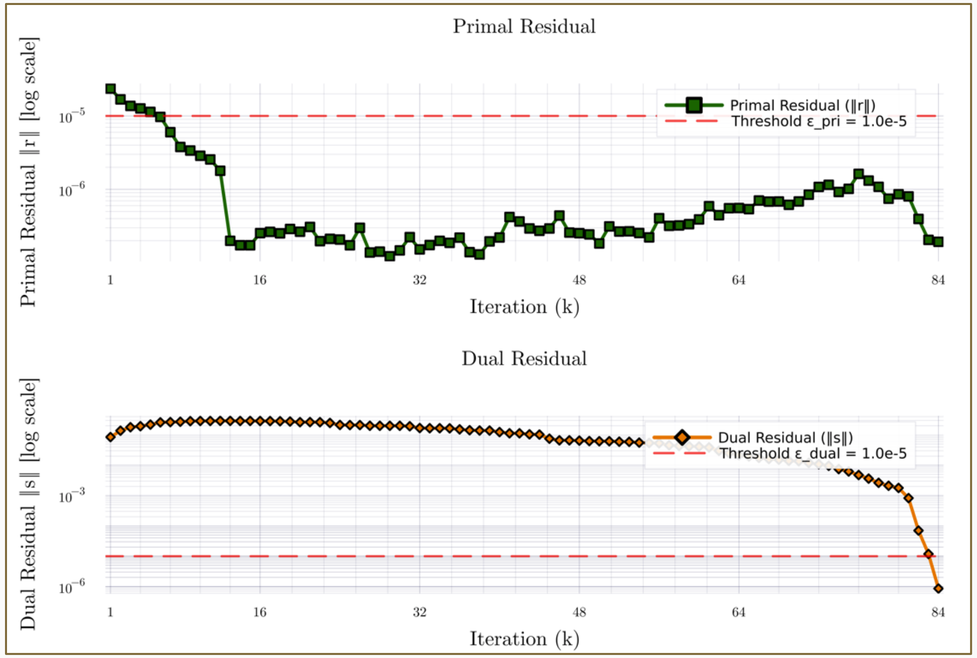
Brute Forced PV Action Plots

tADMM PV Action Plots

tADMM – temporal ADMM

MPOPF Simulation results for IEEE123A_1ph – Convergence



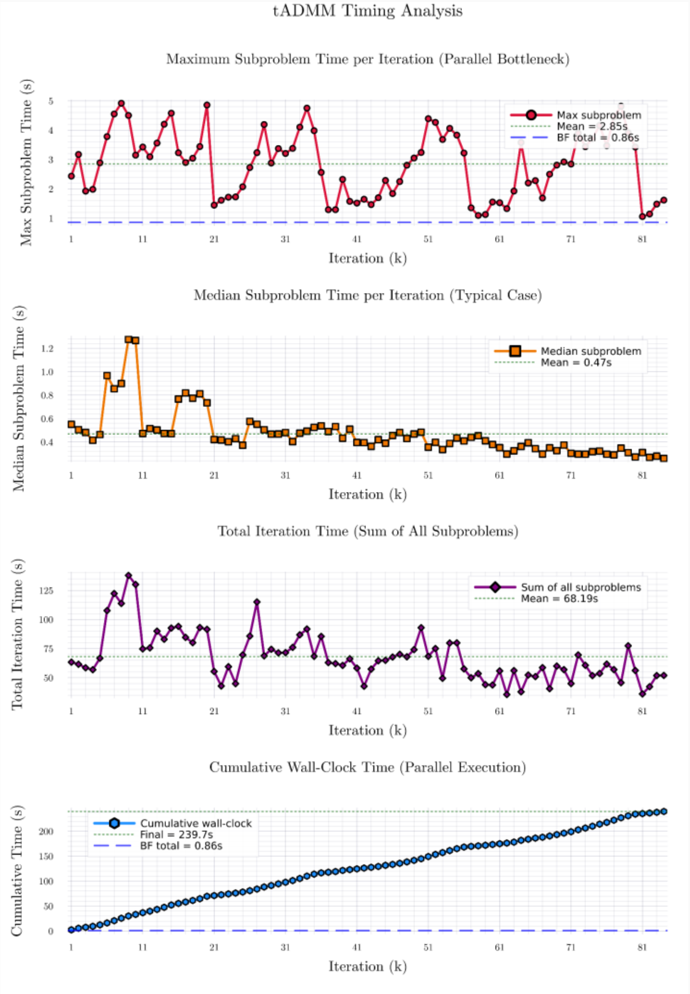


Convergence of Objective Function:
tADMM (Blue) vs **Brute Forced** (Orange)

Convergence of norms:
r-norm (Green) vs **s-norm** (Orange)

tADMM – temporal ADMM

A temporal decomposition algorithm for MPOPF



Currently tADMM is converging slower than Brute-forced optimization which shouldn't be the case since typically solution times of nonlinear problems scale superlinearly with problem size.

We feel that this might either be due to some poor optimization or some other bugs.

Currently under investigation

Future Work and Timeline

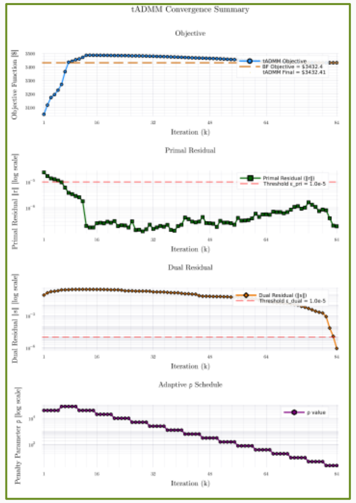
- Perfecting Temporal Decomposition for Medium Sized Balanced Three-Phase Systems
- Temporal Decomposition for Large Sized Unbalanced Three-Phase Systems
- Concluding Research and Dissertation
- Timeline Gantt Chart

Future Work and Timeline

Task 1: Temporal Decomposition for Medium 1ph Systems

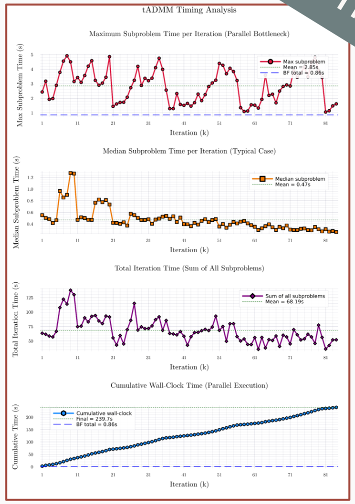
- Already successfully achieved: IEEE 123 Balanced-Three Phase Network + 17 PVs + 26 Batteries for $T = 96$ timesteps ($24h$ at $\Delta t = 15\text{min}$) solved to optimality using tADMM
- But current implementation is somehow slower in time compared to brute force solution, perhaps due to poorly optimized code or other reasons – currently in investigation

➢Now to December 2025



Good convergence

To fix!



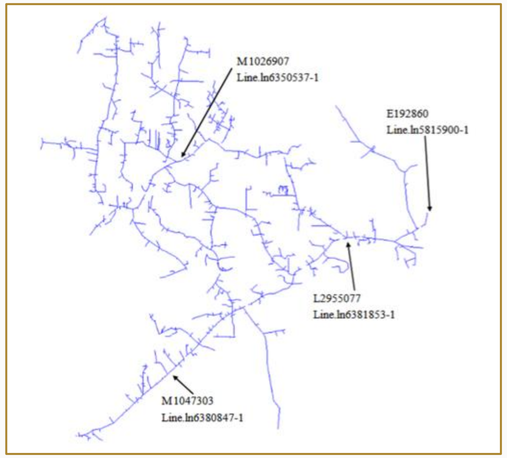
Slow optimization

Future Work and Timeline

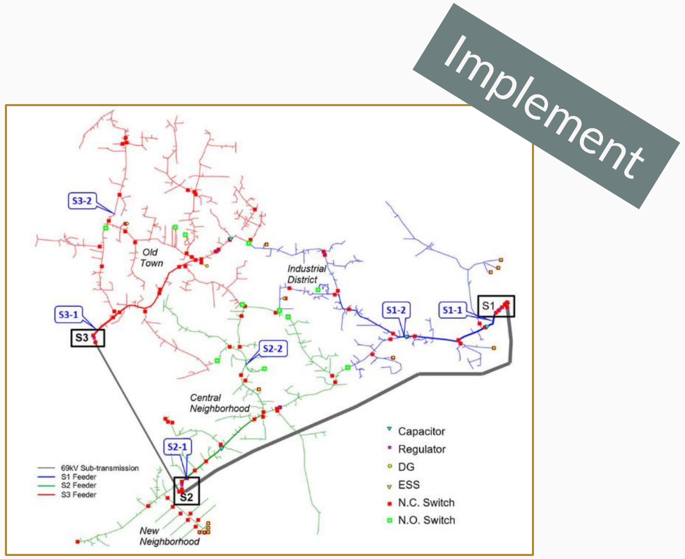
Task 2: Temporal Decomposition for Large Unbalanced 3ph Systems

- Formulate and implement tADMM for larger test systems like IEEE8500 3ph and 9500 3ph systems
  - First start with LinDistFlow3ph implementation, and if successful attempt to extend to nonlinear 3ph formulation

➢ Jan 2026 to March 2026



IEEE 8500 3ph System [ieee8500]



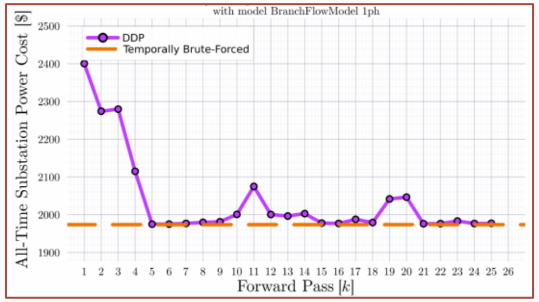
9500 3ph System [9500Test]

Future Work and Timeline

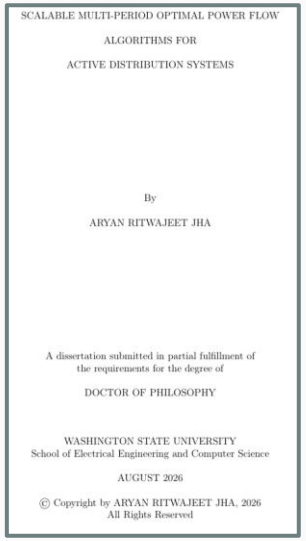
Task 3: Concluding Research and Dissertation

- Complete investigating DDP and document its performance in a paper
- Conduct a final comprehensive literature review on novel temporal decomposition methods for MPOPF
- Complete any remaining implementations or case studies

➢Feb 2026 to May 2026



Try to finalize conclusion about DDP performance for MPOPF



SCALABLE MULTI-PERIOD OPTIMAL POWER FLOW

ALGORITHMS FOR

ACTIVE DISTRIBUTION SYSTEMS

By

ARYAN RITWAJEET JHA

A dissertation submitted in partial fulfillment of
the requirements for the degree of

DOCTOR OF PHILOSOPHY

WASHINGTON STATE UNIVERSITY
School of Electrical Engineering and Computer Science

AUGUST 2026

© Copyright by ARYAN RITWAJEET JHA, 2026
All Rights Reserved

Finish writing dissertation

Future Work and Timeline

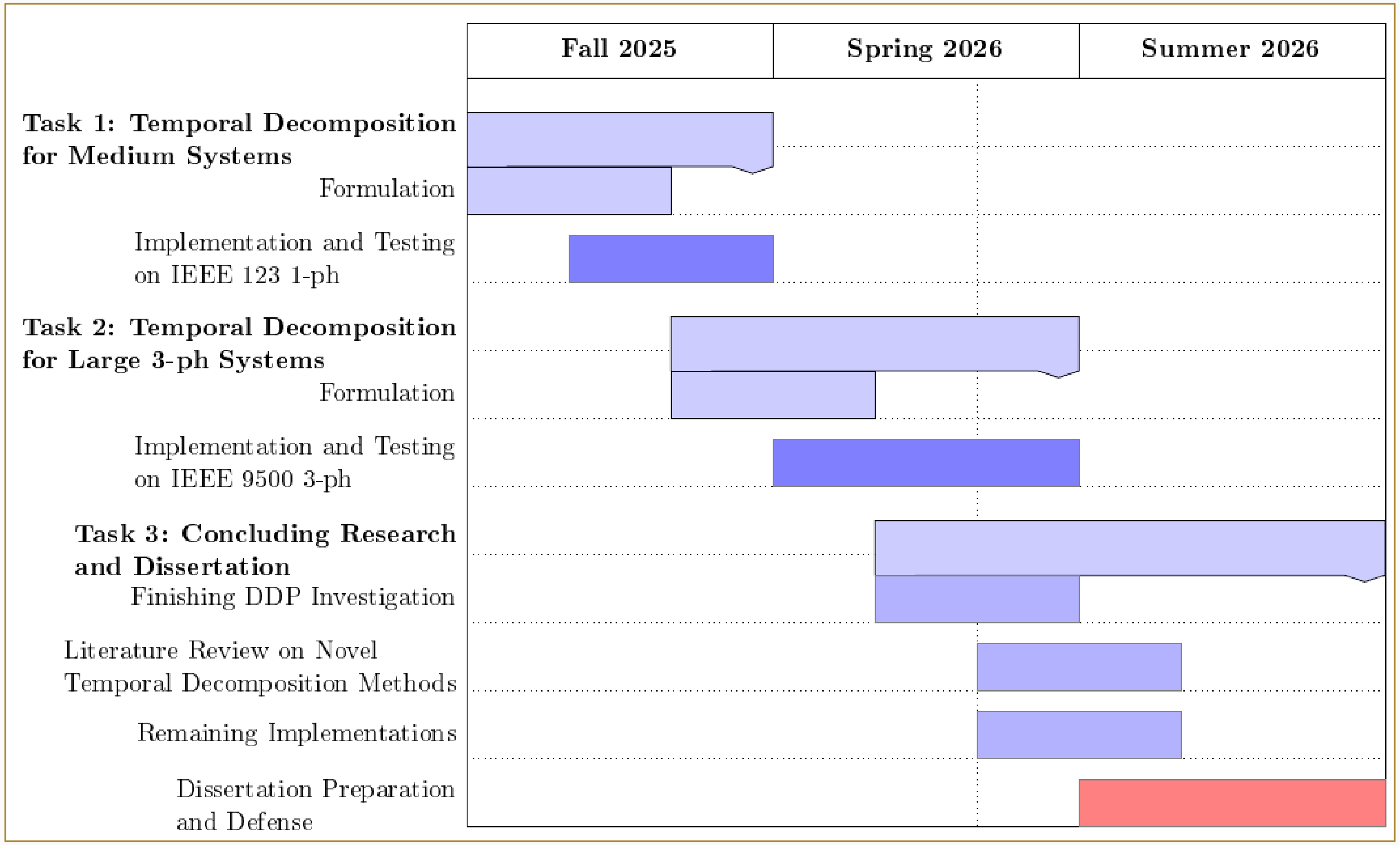
Task 4: Multi Source Optimal Power Flow

- Mention work with Chandra Kant!

➢ Nov 2025

Future Work and Timeline

Timeline as Gantt Chart



Gantt chart showing execution plan for future works

References

References

1. [9500Test] Introducing the 9500 Node Distribution Test System to Support Advanced Power Applications. (2023, February 03). Retrieved from https://www.pnnl.gov/publications/introducing-9500-node-distribution-test-system-support-advanced-power-applications
2. [ieee8500] Arritt, R. F., & Dugan, R. C. . The IEEE 8500-node test feeder. IEEE PES T&D 2010. IEEE. doi: 10.1109/TDC.2010.5484381