Learning to Solve Differential Equation Constrained Optimization Problems

Vincenzo Di Vito *, Mostafa Mohammadian ^, Kyri Baker ^, Ferdinando Fioretto *

* Department of Computer Science, University Of Virginia
^ College of Engineering and Applied Science, University Of Colorado Boulder

University of Virginia

University of Colorado **Boulder**

Overview

- Motivations
- Problem setting
- Challenges
- Proposed approach
- Experimental setting/results
- Conclusions

Motivations

- Many decision-making process interacts with dynamic phenomena, which are governed by system of differential equations.
- Optimizing the decision variables while simultaneously solving the associated DEs poses significant computational challenges.
- Classical optimization-based approach struggle with scalability, efficiency and non linear components, often disregarding the dynamic behaviors of the system.

Problem setting

$$\underset{\boldsymbol{u}}{\text{Minimize}} \quad \overbrace{L(\boldsymbol{u}, \boldsymbol{y}(T)) + \int_{t=0}^{T} \Phi(\boldsymbol{u}, \boldsymbol{y}(t), t)\, dt}^{\mathcal{J}(\boldsymbol{u}, \boldsymbol{y}(t))} \tag{1a}$$

$$\texttt{s.t.} \quad d\boldsymbol{y}(t) = \boldsymbol{F}(\boldsymbol{u}, \boldsymbol{y}(t), t)dt \tag{1b}$$

$$\boldsymbol{y}(0) = \boldsymbol{I}(\boldsymbol{u}) \tag{1c}$$

$$\boldsymbol{g}(\boldsymbol{u}, \boldsymbol{y}(t)) \leq 0; \quad \boldsymbol{h}(\boldsymbol{u}, \boldsymbol{y}(t)) = 0, \tag{1d}$$

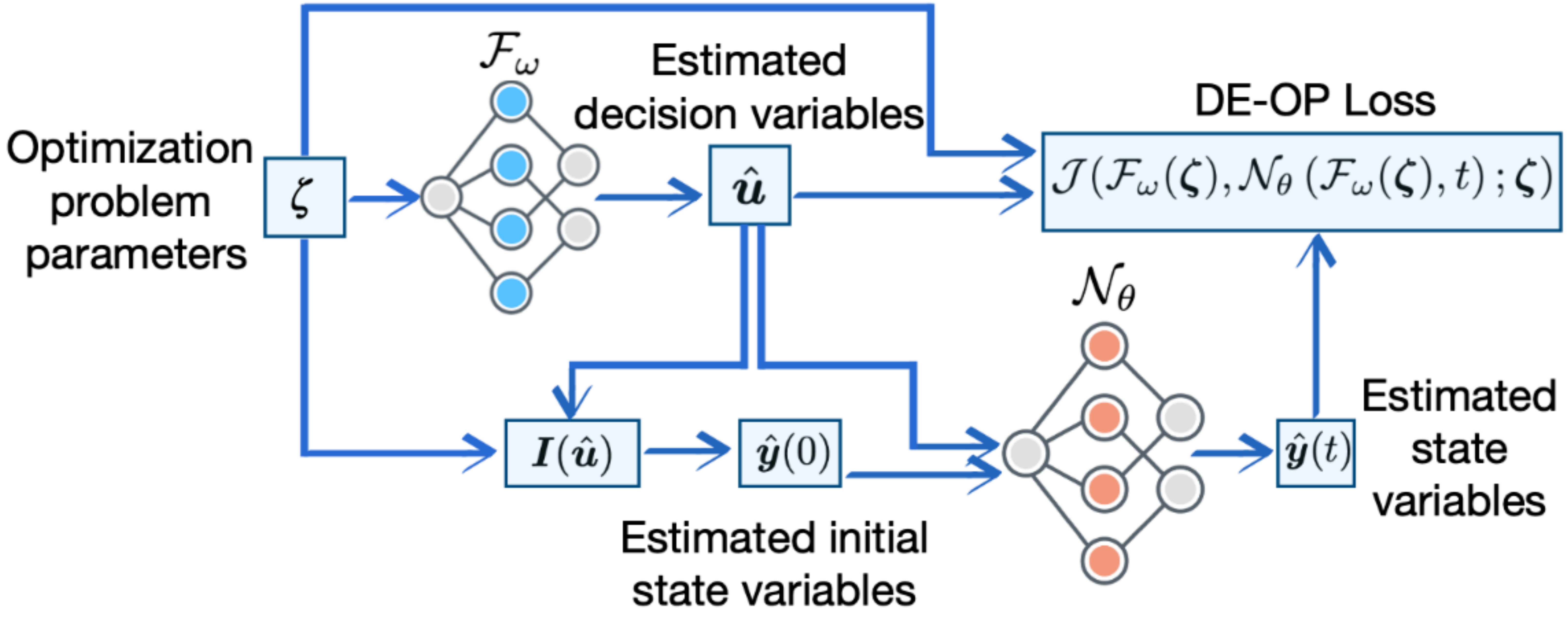
Decision variables

State variables

Proposed approach

Differential-Equation Optimization Proxy (DE-OP):

A dual network with a Learning to Optimize model $\mathcal{F}_\omega$ to approximate the decision variables and a Neural-differential equation model $\mathcal{N}_\theta$ to capture the system dynamics.



Proposed approach

Learning task

$$\operatorname*{Minimize}_{\omega,\theta} \mathbb{E}_{\zeta\sim\Pi} \left[\mathcal{J}\left(\mathcal{F}_\omega(\zeta), \mathcal{N}_\theta(\mathcal{F}_\omega(\zeta), t); \zeta\right)\right] \tag{2a}$$

$$\texttt{s.t.} \quad \text{(1b)–(1d)}, \tag{2b}$$

State variables estimates

$$d\hat{\boldsymbol{y}}(t) = \mathcal{N}_\theta(\hat{\boldsymbol{u}}, t)dt \tag{3a}$$

$$\hat{\boldsymbol{y}}(0) = \boldsymbol{I}(\hat{\boldsymbol{u}}). \tag{3b}$$

Neural-DE model initialization

Given that the neural-DE model takes as input the LtO's estimated decisions, they can be initialized on steady-state decisions, to provide accurate estimate of the state variables:

$$\operatorname*{Minimize}_{\theta} \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}} \left[\|\mathcal{N}_\theta(\boldsymbol{x}, t) - \boldsymbol{y}(t)\|^2\right], \tag{4}$$

Proposed approach

Loss function

$$\mathcal{L}^{\text{DE-OP}}(\hat{\boldsymbol{u}}, \boldsymbol{u}^\star, \hat{\boldsymbol{y}}(t)) = \|\hat{\boldsymbol{u}} - \boldsymbol{u}^\star\|^2 + \boldsymbol{\lambda}_{h'}^\top |\boldsymbol{h}'(\hat{\boldsymbol{u}}, \hat{\boldsymbol{y}}(t))| + \boldsymbol{\lambda}_g^\top \max(0, \boldsymbol{g}(\hat{\boldsymbol{u}}, \hat{\boldsymbol{y}}(t))), \tag{6}$$

Training algorithm: Primal-Dual learning

Algorithm 1 Primal Dual Learning for DE-Constrained Optimization

Input: Dataset $\mathcal{D} = \{(\boldsymbol{\zeta}_i, \boldsymbol{u}_i^\star)\}_{i=1}^N$; optimizer method, learning rate η and Lagrange step size ρ.
Initialize Lagrange multipliers $\boldsymbol{\lambda}_{h'}^0 = 0$, $\boldsymbol{\lambda}_g^0 = 0$.
For each epoch $k = 0, 1, 2, \ldots$
    For each $(\boldsymbol{\zeta}_i, \boldsymbol{u}_i^\star) \in \mathcal{D}$
        $\hat{\boldsymbol{u}}_i \leftarrow \mathcal{F}_{\omega^k}(\boldsymbol{\zeta}_i), \quad \hat{\boldsymbol{y}}_i(t) \leftarrow \mathcal{N}_{\theta^k}(\mathcal{F}_{\omega^k}(\boldsymbol{\zeta}_i), t)$
        Compute loss function: $\mathcal{L}^{\text{DE-OP}}(\hat{\boldsymbol{u}}_i, \boldsymbol{u}_i^\star, \hat{\boldsymbol{y}}_i(t))$ using (6)
        Update DE-OP model parameters:

$$\omega^{k+1} \leftarrow \omega^k - \eta \nabla_\omega \mathcal{L}^{\text{DE-OP}}\left(\mathcal{F}_{\omega^k}^{\boldsymbol{\lambda}^k}(\boldsymbol{\zeta}), \boldsymbol{u}^\star, \mathcal{N}_{\theta^k}^{\boldsymbol{\lambda}^k}\left(\mathcal{F}_{\omega^k}^{\boldsymbol{\lambda}^k}(\boldsymbol{\zeta}), t\right)\right)$$

$$\theta^{k+1} \leftarrow \theta^k - \eta \nabla_\theta \mathcal{L}^{\text{DE-OP}}\left(\mathcal{F}_{\omega^k}^{\boldsymbol{\lambda}^k}(\boldsymbol{\zeta}), \boldsymbol{u}^\star, \mathcal{N}_{\theta^k}^{\boldsymbol{\lambda}^k}\left(\mathcal{F}_{\omega^k}^{\boldsymbol{\lambda}^k}(\boldsymbol{\zeta}), t\right)\right)$$

    Update Lagrange multipliers:

$$\boldsymbol{\lambda}_{h'}^{k+1} \leftarrow \boldsymbol{\lambda}_{h'}^k + \rho|\boldsymbol{h}'(\hat{\boldsymbol{u}}, \hat{\boldsymbol{y}}(t))|, \qquad \boldsymbol{\lambda}_g^{k+1} \leftarrow \boldsymbol{\lambda}_g^k + \rho \max(0, \boldsymbol{g}(\hat{\boldsymbol{u}}, \hat{\boldsymbol{y}}(t))).$$

Experimental setting

Model 2 The Stability Constrained AC-OPF Problem

$$
\begin{aligned}
&\textbf{Parameters}: \quad \zeta = (\boldsymbol{S}^d)\\
&\textbf{decision variables}: \quad \boldsymbol{u} = (S_i^r, V_i) \ \forall i \in \mathcal{N}, \quad S_{ij} \ \forall (i,j) \in \mathcal{L}\\
&\textbf{State variables}: \quad \boldsymbol{y}(t) = (\boldsymbol{\delta}^g(t), \boldsymbol{\omega}^g(t)) \ \forall g \in \mathcal{G}
\end{aligned}
$$

$$\text{Minimize} \sum_{i \in \mathcal{G}} c_{2i}(\Re(S_i^r))^2 + c_{1i}\Re(S_i^r) + c_{0i} \tag{16a}$$

s. t.

$$(11b) - (11h) \tag{16b}$$

$$\frac{d\delta^g(t)}{dt} = \omega_s(\omega^g(t) - \omega_s) \ \forall g \in \mathcal{G} \tag{16c}$$

$$\frac{d\omega^g(t)}{dt} = \frac{1}{m^g}\left(p_m^g - d^g(\omega^g(t) - \omega_s)\right) - \frac{e_q'^g(0)|V_g|}{x_d'^g m^g}\sin(\delta^g(t) - \theta_g) \ \forall g \in \mathcal{G} \tag{16d}$$

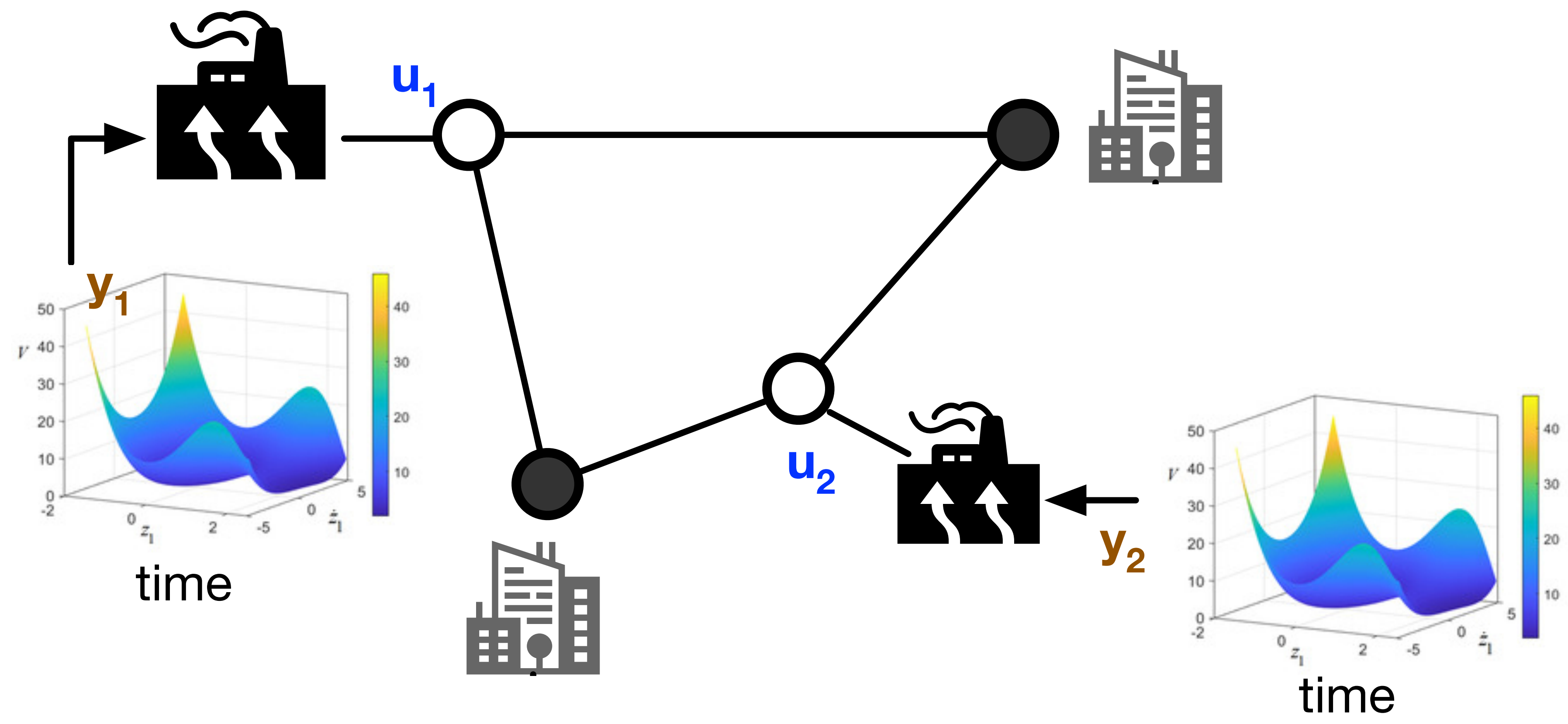
$$\frac{e_q'^g(0)|V_g|\sin(\delta^g(0) - \theta_g)}{x_d'^g} - p_g^r = 0 \ \forall g \in \mathcal{G} \tag{16e}$$

$$\frac{e_q'^g(0)|V_g|\cos(\delta^g(0) - \theta_g) - |V_g|^2}{x_d'^g} - q_g^r = 0 \ \forall g \in \mathcal{G} \tag{16f}$$

$$\omega^g(0) = \omega_s \ \forall g \in \mathcal{G} \tag{16g}$$

$$\delta^g(t) \le \delta^{\max} \ \forall g \in \mathcal{G}. \tag{16h}$$

(16i)



Steady-state constraints

$$v_i^l \le |V_i| \le v_i^u \ \ \forall i \in N \tag{11b}$$

$$-\theta_{ij}^{\Delta} \le \angle(V_i V_j^*) \le \theta_{ij}^{\Delta} \ \ \forall (i,j) \in \mathcal{L} \tag{11c}$$

$$S_i^{rl} \le S_i^r \le S_i^{ru} \ \ \forall i \in \mathcal{N} \tag{11d}$$

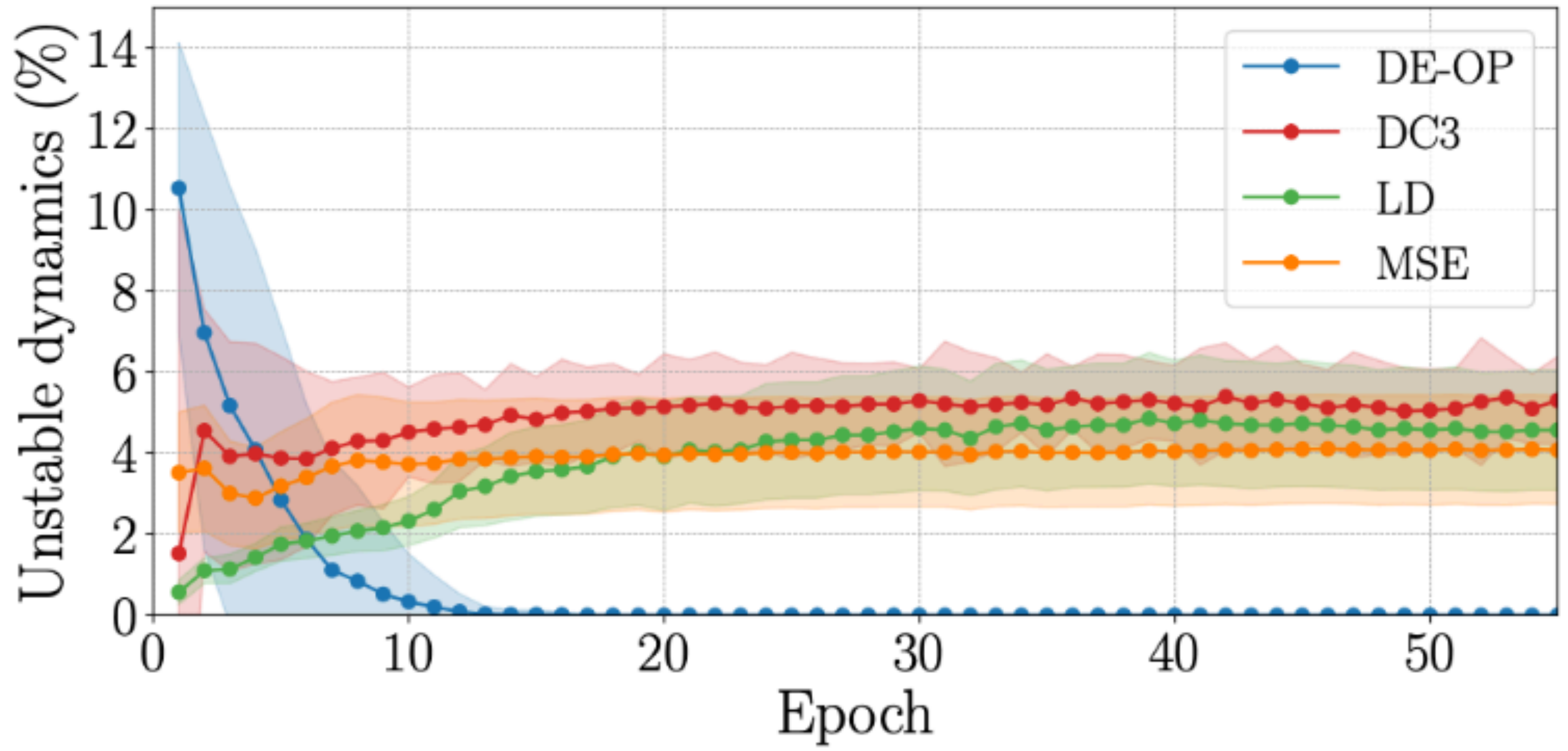
$$|S_{ij}| \le s_{ij}^u \ \ \forall (i,j) \in \mathcal{L} \tag{11e}$$

$$S_i^r - S_i^d = \textstyle\sum_{(i,j)\in L} S_{ij} \ \ \forall i \in \mathcal{N} \tag{11f}$$

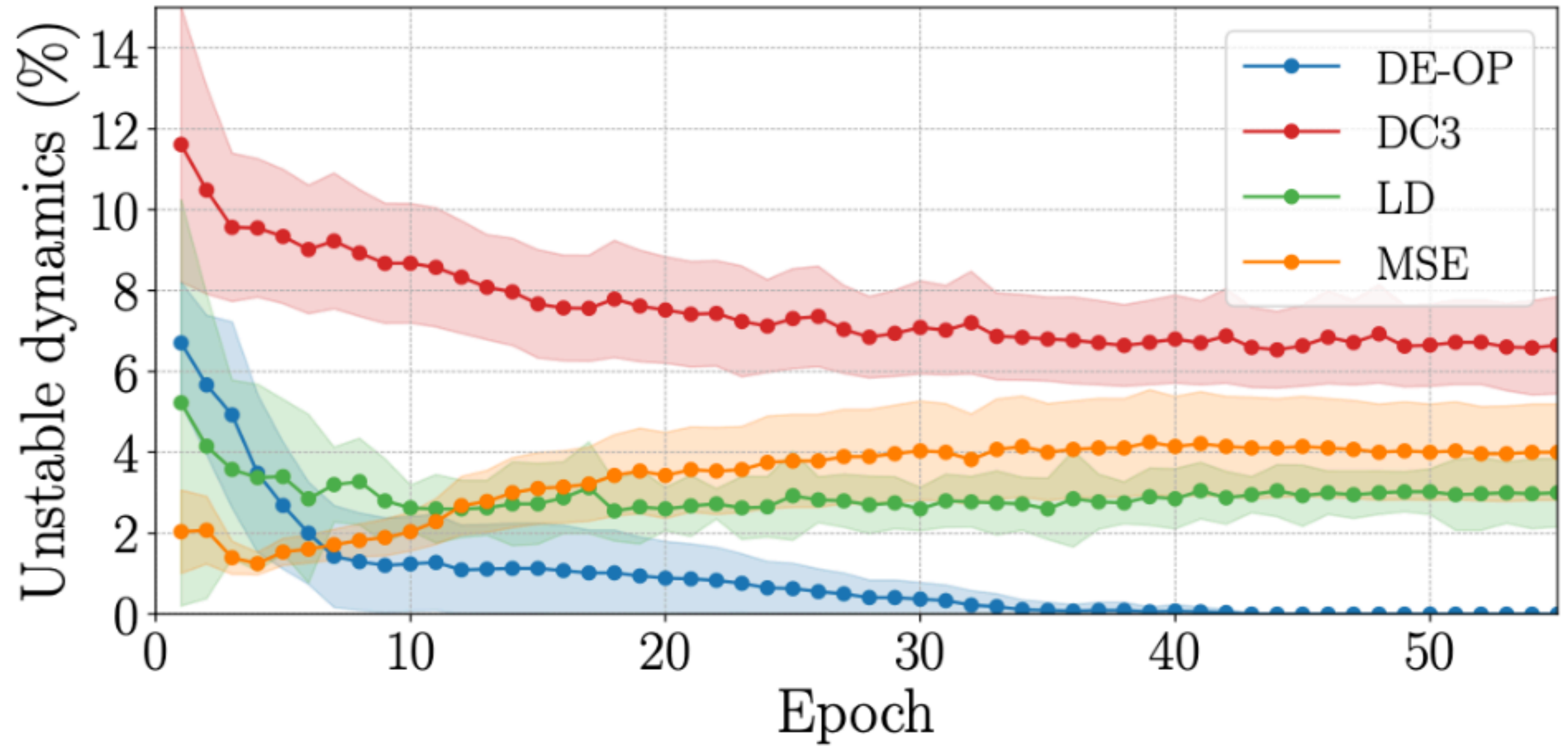
$$S_{ij} = Y_{ij}^*|V_i|^2 - Y_{ij}^* V_i V_j^* \ \ \forall (i,j) \in \mathcal{L} \tag{11g}$$

$$\theta_{\text{ref}} = 0 \tag{11h}$$

Experimental results



WSCC-9 bus system: Percentage of unstable dynamics at training time for different approaches



IEEE-57 bus system: Percentage of unstable dynamics at training time for different approaches

Models		Metrics		
$\mathcal{F}_\omega$	$\mathcal{N}_\theta$	Stability Vio.	Flow Vio. $\times 10^{-3}$	Boundary Vio. $\times 10^{-4}$
DE-OP (ours)		**0.00**	9.15 ± 0.442	0.25 ± 0.172
MSE	$\emptyset$	23.30 ± 0.206	12.65 ± 2.281	6.44 ± 1.434
LD	$\emptyset$	23.10 ± 0.219	6.23 ± 0.125	0.00
DC3	$\emptyset$	28.60 ± 0.232	0.00	0.00

Test-set constraint violations

Models		WSCC 9-bus	IEEE 57-bus
$\mathcal{F}_\omega$	$\mathcal{N}_\theta$	Inference Time (sec)	
DE-OP (ours)		0.001 ± 0.00	0.009 ± 0.00
MSE	$\emptyset$	0.000 ± 0.00	0.001 ± 0.00
LD	$\emptyset$	0.000 ± 0.00	0.001 ± 0.00
DC3	$\emptyset$	0.025 ± 0.00	0.089 ± 0.00

Conclusion

- Motivated by the complexity and computational requirements of DE-constrained optimization problems, we proposed DE-OP, a novel learning-based method.
- DE-OP consists of a dual network architecture, where a Learning to Optimize model approximates the decision variables and a neural-DE model approximates the state variables.
- DE-OP is trained adopting a primal-dual learning approach, where estimates of decision, state, and dual variables are iteratively refined to satisfy system dynamics and constraints.
- Experimental results demonstrate that DE-OP can produce near optimal solutions in real-time, while adhering dynamic constraints.
- Future work will focus on extending this idea to a broader class of DE-constrained optimization problems and physics-informed learning frameworks.