Distributed Optimization and Distributed Learning: A Paradigm Shift for Power Systems

Ahmad Al-Tawaha*§,Elson Cibaku†§,SangWoo Park†,Javad Lavaei‡,and Ming Jin*

***Abstract*—This survey provides a comprehensive overview of recent advances in distributed optimization and machine learning for power systems, particularly focusing on optimal power flow (OPF) problems. We cover distributed algorithms for convex relaxations and nonconvex optimization, highlighting key algorithmic ingredients and practical considerations for their implementation. Furthermore, we explore the emerging field of distributed machine learning, including deep learning and (multi-agent) reinforcement learning, and their applications in areas such as OPF and voltage control. We investigate the synergy between optimization and learning, particularly in the context of learning-assisted distributed optimization, and provide the first comprehensive survey of distributed real-time OPF, addressing time-varying conditions and constraint handling. Throughout the survey, we emphasize practical considerations such as data efficiency, scalability, and safety, aiming to guide researchers and practitioners in developing and deploying effective solutions for a more efficient and resilient power grid.**

***Index Terms*—power systems, distributed optimization, machine learning, real-time optimization, nonconvex optimization, reinforcement learning, multi-agent systems, optimal power flow, distributed energy resources**

I. Introduction

The high penetration of distributed energy resources (DERs) introduces increased complexity and uncertainty into power system operation, raising concerns about power quality, voltage issues, stability, privacy, and cybersecurity [1]–[4]. Distributed optimization and learning techniques offer a promising solution by enabling localized decision-making and parallel computation, potentially enhancing efficiency, data privacy, real-time response, and resilience to cyberattacks and component failures, while also alleviating communication bottlenecks. Machine learning (ML), including deep learning (DL) and reinforcement learning (RL), has emerged as a powerful tool for handling nonlinearities and uncertainties inherent in energy grids [5]. The synergy between distributed optimization and ML, particularly in distributed learning techniques, holds the potential to revolutionize power system operations.

This survey aims to be a valuable resource for researchers and practitioners across power systems, control theory, optimization, and ML, offering insights into the application of distributed optimization and distributed ML techniques to power system problems, particularly optimal power flow (OPF). Building upon the foundation laid by existing comprehensive surveys on distributed optimization (e.g., [1]–[3], [6]) and ML (e.g., [5]), we delve into the intersection of these fields with a focus on the following key aspects. Specifically, the reader would benefit from:

- A unified agent-based decomposition framework (Sec. II) as a pedagogical tool to aid newcomers in understanding initial formulations of various distributed optimization algorithms for different power flow models (Sec. III).
- Recent progress in addressing nonconvex problems (Sec. III), focusing on methods with practical benefits for power systems, such as *ADMM variants* (modified forms of the classical ADMM—multi-block, proximal, asynchronous, or accelerated—that lower per-iteration cost and tolerate non-ideal communication), ALADIN, and other promising approaches (e.g., distributed interior-point methods).
- ML applications to distributed OPF, including distributed DL/RL (Secs. IV-A and IV-B), multi-agent RL (MARL) (Sec. IV-C), and the synergy between optimization and ML in learning-assisted distributed optimization (Sec. IV-D) i.e., distributed optimizers that incorporate ML models into their update rules to improve convergence or solution quality.
- The first comprehensive survey of distributed real-time OPF (RT-OPF) (Sec. V), highlighting connections to decomposition techniques (Sec. III), the role of real-time measurements in algorithm design, and challenges/advances in handling time-varying conditions and constraints. We also explore potential cross-pollination with low per-iteration cost algorithms such as ADMM.

Throughout this survey, we emphasize the practical challenges of applying distributed optimization and ML techniques to power systems. We highlight how recent research addresses these challenges in various contexts, such as ensuring data efficiency and safety in DL for OPF (Sec. IV-A), tackling the issues of nonstationarity, partial observability, and communication efficiency in MARL (Sec. IV-C), and guaranteeing safety and stability constraints in distributed RT-OPF (Sec. V). By distilling key ideas from recent research advances, we aim to provide guidance for both researchers and practitioners. We conclude by identifying key challenges and future research directions (Sec. VI) to inspire further innovation in the field.

While this survey offers a comprehensive overview of the key areas outlined above, it does not cover all aspects of distributed optimization and ML for power systems. We

§ Co-first authors.
*Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA.
†Department of Mechanical and Industrial Engineering, New Jersey Institute of Technology, Newark, NJ, USA.
‡Department of Industrial Engineering and Operations Research, University of California, Berkeley, Berkeley, CA, USA.

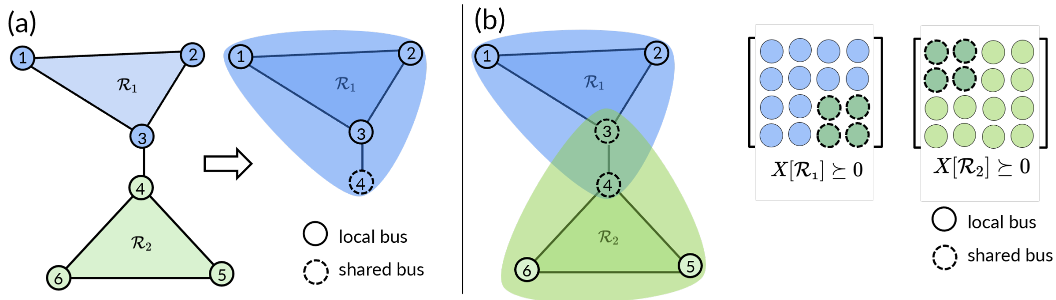

Fig. 1. Distributed SDP relaxation for 6-bus power network. **(a)** Network partitioned into regions $\mathcal{R}_1$ (blue, buses 1-3) and $\mathcal{R}_2$ (green, buses 4-6), with boundary buses 3 and 4 and decomposed view showing $\mathcal{R}_1$ with local buses $1, 2, 3$ plus shared bus 4, and $\mathcal{R}_2$ with local buses $4, 5, 6$ plus shared bus 3. **(b)** Left: Overlapping graph showing shared buses 3 and 4. Right: Decomposition of global PSD matrix $X \succeq 0$ into submatrices $X[\mathcal{R}_1] \succeq 0$ and $X[\mathcal{R}_2] \succeq 0$, with consensus constraints ensuring agreement on shared variables.

refer readers to existing surveys for technical details on the original OPF formulation and its convex relaxations (e.g., [1, Sec. II-B], [7, Sec. II]), specific problem formulations for distribution systems (e.g., volt/var control and retail markets) [3], distributed optimization for discrete decision variables [6], and centralized RT-OPF methods [8].

II. A Unified Distributed Formulation of OPF

A. Agent-based Decomposition and Consensus Formulation

Consider a power network represented as a graph $\mathcal{G}(\mathcal{N}, \mathcal{L})$, where $\mathcal{N}$ denotes the set of buses, and $\mathcal{L}$ represents the set of transmission lines connecting these buses. To enable distributed optimization, we partition the network into K regions $\mathcal{R}_1, \ldots, \mathcal{R}_K$, each managed by a separate agent, as illustrated in Fig. 1(a).

In this decomposition, each agent is responsible for its local region and maintains local variables corresponding to the electrical states of the buses within that region. To ensure consistency at the boundaries between regions, agents also maintain copies of the boundary variables from neighboring regions. These copies are virtual constructs used solely for computational purposes to facilitate coordination among agents; they do not represent any physical components in the power system.

For each region $\mathcal{R}_k$, we define $\mathcal{N}_k^{\text{local}} \subseteq \mathcal{N}$ as the set of local buses within region k. The set $\mathcal{N}_k^{\text{shared}} \subseteq \mathcal{N}$ consists of buses that are at the boundaries of region k and are shared with neighboring regions. The decision variables for region $\mathcal{R}_k$ are denoted as $x_k = (x_k^{\text{local}}, x_k^{\text{shared}})$, where x_k^{local} represents the electrical states of local buses, including variables such as voltage angles, voltage magnitudes, active power injections, and reactive power injections. The vector x_k^{shared} consists of the copies of the electrical states of the shared buses from neighboring regions. By maintaining these copies, agents can coordinate with their neighbors to ensure that the values of shared variables are consistent across the network. We use $x_k[i]$ to denote the i-th entry of x_k, $X[i, j]$ for the entry at the i-th row and j-th column of X, and $X[I]$ for the principal submatrix of X indexed by I.

The distributed OPF problem can be formulated as:

$$\min_{\{x_k\}_{k=1}^K} \quad \sum_{k=1}^K f_k(x_k) \tag{1a}$$

$$\text{s.t.} \quad \sum_{k=1}^K A_k x_k = b_k \tag{1b}$$

$$x_k \in \mathcal{X}_k \quad \forall k \in \{1, \ldots, K\} \tag{1c}$$

where $f_k(x_k)$ is the local cost function for region $\mathcal{R}_k$. The consensus constraint (Eq. 1b) ensures consistency in the shared variables between regions. The coupling matrix A_k associated with region $\mathcal{R}_k$ encodes the consensus constraints for all shared buses in $\mathcal{R}_k$. This constraint enforces equality of electrical states (such as voltage angles and magnitudes) at the shared buses across neighboring regions, effectively coupling the subproblems while allowing for parallel computation.

The local feasibility set $\mathcal{X}_k$ for region $\mathcal{R}_k$ is defined as: $\mathcal{X}_k = \{x_k : h_k(x_k) = 0, g_k(x_k) \leq 0\}$ where $h_k(x_k) = 0$ represents the nonlinear equality constraints, including power flow equations within the region, and $g_k(x_k) \leq 0$ represents the inequality constraints, encompassing operational limits on physical variables such as voltage magnitudes, power generation limits, and line flow constraints. This formulation allows each region to handle its local constraints independently while maintaining system-wide consistency through the consensus constraints.[1]

1) Distributed Nonconvex Formulation: In *region-based decomposition* for AC OPF (e.g., [9]), the nonconvex nature of the AC OPF problem arises from the nonlinear power flow equations represented by $h_k(x_k) = 0$. An alternative approach, known as *component-based decomposition* [10], [11], considers each network element (generators, transformers, loads, transmission lines) as an individual agent. This method offers a finer granularity in problem decomposition compared to the region-based approach.

2) Distributed SDP Relaxation: The semidefinite programming (SDP) relaxation of the OPF problem introduces a matrix variable $X \in \mathbb{H}^{|\mathcal{N}|}$ (where $\mathbb{H}$ is the set of $n \times n$ Hermitian

[1]While this consensus-based formulation is general, certain problem structures, such as TSO-DSO coordination, are often better captured by alternative frameworks like hierarchical or bi-level optimization, discussed in MARL (Sec. IV-C) and RT-OPF (Sec. V) contexts.

matrices and $|\mathcal{N}|$ is the number of buses) to represent the outer product of the voltage phasors, i.e., $X = UU^*$, where $U \in \mathbb{C}^{|\mathcal{N}|}$ is the vector of complex numbers of size $|\mathcal{N}|$. The matrix X is symmetric and positive semidefinite by construction. The SDP relaxation drops the rank-1 constraint on X, replacing it with the PSD constraint $X \succeq 0$. For the distributed formulation, each region can correspond to a "bag" of nodes in a tree/chordal/clique decomposition of $\mathcal{G}$ [12] (as illustrated in Fig. 1(b)). For each agent $k \in \mathcal{K}$, the local variable x_k is the principal submatrix $X[\mathcal{R}_k] \in \mathbb{H}^{|\mathcal{R}_k|}$ of X, indexed by the buses in subregion $\mathcal{R}_k$. Within x_k, the consensus variables are the entries corresponding to the buses that appear in neighboring bags of the tree decomposition, i.e., separator sets. The consensus constraint (Eq. 1b) ensures consistency of these entries across bags (as shown in Fig. 1(b)). If the optimal solution to the SDP relaxation is rank-1, it solves the original OPF problem, i.e., the relaxation is exact. In practice, even with higher-rank solutions, near-optimal solutions to the OPF can often be constructed. The local constraint set $\mathcal{X}_k$, with the addition of the PSD constraint $X[\mathcal{R}_k] \succeq 0$, is convex under the SDP relaxation.

3) Distributed SOCP Relaxation: In second-order cone programming (SOCP) relaxation, power flow equations use branch power flows and voltage magnitudes, derived from SDP relaxation by imposing additional constraints on X while removing the $X \succeq 0$ constraint. SOCP constraints have a simpler structure and can be handled by specialized solvers. For radial networks, the SOCP relaxation achieves exactness under mild conditions [13]. In this topology, each bus acts as an agent with local variable x_k, encompassing squared voltage magnitude, net complex power injection, branch power flow, squared branch current magnitude to its ancestor, and local copies of neighbor variables. The local constraint set $\mathcal{X}_k$ includes the SOCP constraints.

4) Communication Topology and Protocol: Communication topology, represented as a directed/undirected graph, determines agent information exchange patterns. Common structures include m-hop neighbor distributed networks, star networks with central coordination, and hierarchical networks with level-based communication. The information exchange protocol specifies shared data types, frequency, and timing (e.g., primal/dual variables), incorporating synchronous/asynchronous updates, event-triggered schemes, and time-varying graphs [2, Sec. 4].

III. Nonconvex Distributed Optimization Techniques

A. ADMM and Variants

The distributed OPF formulation (Eq. 1) can be interpreted as a multi-block extension of the classical two-block ADMM [14], extending it to handle K blocks of variables $\{x_k\}_{k\in\mathcal{K}}$ with separable objectives $\{f_k\}_{k\in\mathcal{K}}$ and consensus constraint (Eq. 1b). While ADMM is naturally suited to address both local and consensus constraints, its direct multi-block extension may fail to converge [15], necessitating further modifications.

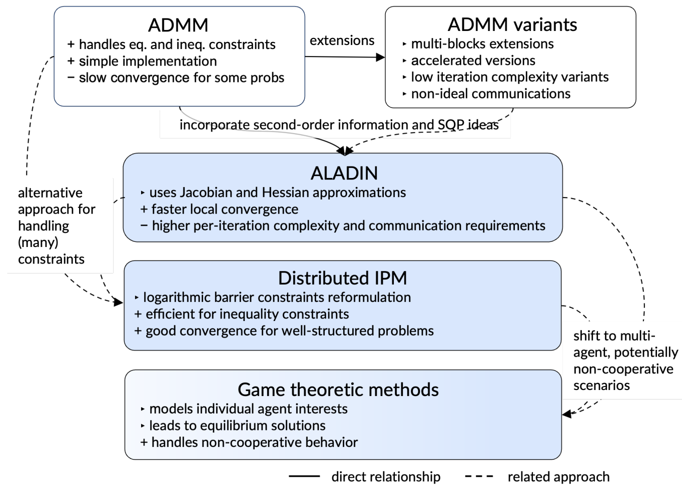

Fig. 2. Distributed optimization method relationships, color-coded by information use: first-order (white), second-order (blue), and hybrid approaches (transition blue).

Problem 1 is suitable for *primal decomposition*, since fixing coupling variables decouples the problem into subproblems. Thereby, it can be reformulated as a two-block problem:

$$\begin{aligned} \min_{x,z} \quad & \sum_{k\in\mathcal{A}} f_k(x_k) \\ \text{s.t.} \quad & A_k x_k - z_k = b_k,\ x_k \in \mathcal{X}_k, \quad \forall k \in \mathcal{K}, \\ & \textstyle\sum_{k\in\mathcal{K}} z_k = 0, \end{aligned} \tag{2}$$

where we introduce the auxiliary variables $z = \{z_k\}_{k\in\mathcal{K}}$ to facilitate consensus update. This allows exploiting classical ADMM on the blocks x and z, with the primal update of x naturally decomposing across agents $k \in \mathcal{K}$. Also, the slack z update admits a closed-form solution, i.e., $z_k^{t+1} = A_k x_k^{t+1} - b_k - d^{t+1}$, for all $k \in \mathcal{K}$, where $d^{t+1} = \frac{1}{K}\left(\sum_{k\in\mathcal{K}} A_k x_k^{t+1} - b\right)$ is the average violation of the coupling constraint (Eq. 1b). The primal update is:

$$\begin{aligned} x_k^{t+1} = \arg\min_{x_k\in\mathcal{X}_k} \quad & f_k(x_k) + (\lambda^t)^\top (A_k x_k) \\ & + \frac{\rho}{2}\|A_k x_k - A_k x_k^t + d^t\|^2, \quad \forall k \in \mathcal{K}, \end{aligned}$$

which can be performed in parallel by agents, and the dual update is: $\lambda^{t+1} = \lambda^t + \rho d^{t+1}$, where λ_k are the dual variables associated with (Eq. 1b) and $\rho > 0$ is a penalty parameter. Problem 1 is also amenable to *dual decomposition*, because relaxing the coupling constraint decouples the problem into subproblems. We introduce Lagrange multipliers λ and formulate the Lagrangian function: $L(x,\lambda) = \sum_{k\in\mathcal{K}} f_k(x_k) - \lambda^\top(\sum_{k\in\mathcal{K}} A_k x_k - b)$ and the resulting dual problem: $\min_\lambda \sum_{k\in\mathcal{K}} f_k^*(A_k^\top \lambda) - b^\top \lambda$, where $f_k^*(z) = \sup_{x_k}\{z^\top x_k - f_k(x_k) : x_k \in \mathcal{X}_k\}$ is the Fenchel conjugate of f_k under the assumption of a bounded convex subset $\mathcal{X}_k$. This problem is well known as consensus optimization, where the sum of K objectives is coupled through the consensus variable λ, and can be handled by the classical two-block ADMM via introducing local copies of $\lambda = \{\lambda_k\}_{k\in\mathcal{K}}$

and linking variables $\zeta = \{\zeta_{ij}\}_{(i,j)\in\mathcal{L}}$:

$$\begin{aligned} \min_{\lambda,\zeta} \quad & \sum_{k\in\mathcal{A}} \left(f_k^*(A_k^\top \lambda_k) - \frac{1}{K} b^\top \lambda_k \right) \\ \text{s.t.} \quad & \lambda_i = \zeta_{ij}, \quad \lambda_j = \zeta_{ij}, \quad \forall (i,j) \in \mathcal{L}. \end{aligned} \tag{3}$$

Applying ADMM yields the dual consensus ADMM implementation [16], with the primal updates: $x_k^{t+1} = \arg\min_{x_k\in\mathcal{X}_k} \quad f_k(x_k) + \frac{1}{4\rho|\mathcal{N}_k|}\|A_k x_k - \frac{1}{K}b - w_k^t + \rho\sum_{j\in\mathcal{N}_k}(\lambda_j^t + \lambda_k^t)\|^2$ and $\lambda_k^{t+1} = \frac{1}{2\rho|\mathcal{N}_k|}(A_k x_k^t - \frac{1}{K}b - w_k^t) + \frac{1}{2|\mathcal{N}_k|}\sum_{j\in\mathcal{N}_k}(\lambda_j^t + \lambda_k^t)$, and the dual update $w_k^{t+1} = w_k^t + \rho\sum_{j\in\mathcal{N}_k}(\lambda_k^t - \lambda_j^t)$.

The above variants of multi-block ADMMs, known as parallel ADMM and dual consensus ADMM [16], have been applied to cost allocation in peer-to-peer electricity markets [17], black-start and parallel restoration [18], and distributed OPF [19]. Other multi-block modifications include: 1) *Proximal regularization*: Adds terms of the form $\frac{1}{2}\|x_k - x_k^t\|_{P_k}^2$ to subproblems, with $P_k \succeq 0$. For example, proximal Jacobian ADMM [20] requires $P_k \succ \rho(K-1)A_k^\top A_k$ to be sufficiently large for convergence. Since the proximal coefficient matrices P_k are generally required to be linearly growing with the number of agents K, a slower convergence is likely for larger problems. 2) *Prediction-correction ADMM*: Variants such as Jacobian ADMM with correction step [21] generate a prediction using Jacobian ADMM and then use correction steps for convergence with $\mathcal{O}(1/t)$ iteration complexity. 3) *Block-wise ADMM*: Variants such as [22] artificially split variables into two groups and apply two-block ADMM, often with smaller proximal coefficients than proximal Jacobian ADMM, potentially leading to faster convergence. This is demonstrated in coordinated control for microgrid clusters [23].

1) Accelerated ADMMs: Accelerated ADMMs enhance convergence rates, reducing iterations and communication. Fast ADMM employs Nesterov acceleration, achieving $O(1/t^2)$ convergence for strongly convex problems versus $O(1/t)$ for classical ADMM, though convergence analysis for general problems remains open. Existing techniques [24] modify primal-dual sequences via $\hat{\omega}^{t+1} = \text{acc}(\omega^{t+1}, \omega^t, \omega^{t-1}, \ldots)$, where $\omega^t = \{x^t, \lambda^t\}$ represents primal-dual updates and "acc" denotes acceleration. A guard condition ensuring monotonic decrease of the combined residual is introduced for convergence [24]. The second-order information of the dual function enables Newton-step acceleration of dual updates [25]. Connections between accelerated ADMM variants and continuous-time dynamical systems establish convergence rates through Lyapunov analysis [26]. Moreover, a second-order dynamical system with vanishing damping yields various inertial parameter rules, including Nesterov acceleration, under suitable time discretization [27]. These methods are effective for economic dispatch [28] and AC/DC networks [29].

2) ADMMs with Low Iteration Complexity: ADMM variants reduce per-iteration complexity through approximations and proximal terms (e.g., $\|x - x^t\|^2$ to control approximation accuracy). *Linearized ADMM* uses local linear approximations for simpler updates via projected gradient steps or proximal mappings [30], [31]. For nonconvex problems, bounded primal and dual updates are typically required to construct a sufficiently decreasing and lower bounded Lyapunov function [31]. *Stochastic ADMM* performs gradient-like iterates with noisy gradients of the augmented Lagrangian (AL) function [30], which is useful when explicit functions are unavailable; however, the high variances of stochastic gradients lead to a convergence rate gap: $\mathcal{O}(1/\sqrt{t})$ for stochastic ADMM versus $\mathcal{O}(1/t)$ for its deterministic counterpart. To address this issue, variance reduction techniques have been proposed, including a stochastic path-integrated differential estimator [32], further combined with acceleration techniques [33]. For instance, [34] provides a unified framework for inexact stochastic ADMM covering several well-known algorithms. These variants typically require Lipschitz differentiable objectives and sufficient proximal coefficients to bound inexact update errors.

3) ADMMs with Non-Ideal Communications: Modern power systems are susceptible to random link failures due to network congestion, infrastructure failures, cyber attacks, and privacy-induced noise. ADMM performance in unbalanced distribution networks degrades significantly under high levels of communication failure and noise [19]. Several ADMM variants address this through flexible agent activation mechanisms and asynchronous updates. Asynchronous updates enhance computational efficiency by reducing idle time from delays and packet losses [35].

Formally, an asynchronous ADMM implementation tailored for the distributed optimization formulation in (Eq. 2) proceeds as follows. Each agent k asynchronously updates its local variables according to:

$$\begin{aligned} x_k^{t_k+1} &= \arg\min_{x_k\in\mathcal{X}_k} f_k(x_k) + (\lambda_k^{t_k})^\top (A_k x_k - z_k^{t_k} - b_k) \\ &\quad + \frac{\rho}{2}\|A_k x_k - z_k^{t_k} - b_k\|^2, \\ \lambda_k^{t_k+1} &= \lambda_k^{t_k} + \rho\left(A_k x_k^{t_k+1} - z_k^{t_k} - b_k\right), \end{aligned}$$

and immediately communicates the updated information to neighboring agents. Each agent maintains a local iteration counter t_k and updates the local consensus variable z_k upon receiving updated boundary information from a sufficient subset of its neighbors. (e.g., a fraction p of neighbors, or subject to a maximum delay τ). The update is defined as:

$$\begin{aligned} z_k^{t_k+1} &= \arg\min_{z_k} \left(\lambda_k^{t_k+1}\right)^\top \left(A_k x_k^{t_k+1} - z_k - b_k\right) \\ &\quad + \frac{\rho}{2}\left\|A_k x_k^{t_k+1} - z_k - b_k\right\|^2 + \frac{\alpha}{2}\left\|z_k - z_k^{t_k}\right\|^2. \end{aligned}$$

A master node sets a maximum tolerable delay τ for each worker, proceeding with updates upon receiving sufficient worker responses while enforcing delay bounds. There is often a trade-off between the number of iterations and waiting time, influenced by the delay bound τ and partial synchronization mechanism [35]. Communication problems can be modeled as a time-varying network with asynchronous updates [36]. An asynchronous dual decomposition algorithm has been proposed and compared favorably with existing methods in coordinating DERs under communication asynchrony and computation errors [37]. Additionally, a data server with its own clock cycles to handle asynchronous data exchange for local consensus has been used in [38] to replace the central aggregator in [35], potentially facilitating easier integration into communication networks.

4) Other Considerations: A proximal ADMM variant enables autonomous agent step size selection using only local information, regardless of communication topology [39]. A scaled dual descent AL framework approach handles general nonlinear equality constraints with improved theoretical complexity guarantees [40].

B. Augmented Lagrangian Alternating Direction Inexact Newton (ALADIN)

ALADIN [41] addresses the nonconvex problem (Eq. 1) by solving decoupled problems in primal variables, similar to ADMM, while also requiring an approximation of the constraint Jacobian and Hessian to solve a coupled Quadratic Programming (QP) problem.

Consider local constraints $\mathcal{X}_k$ in (Eq. 1), where h_k and g_k are assumed to be twice continuously differentiable. The key steps involve solving decoupled problems to either local or global optimality for each agent $k \in \mathcal{K}$:

$$\begin{aligned} \min_{x_k} \quad & f_k(x_k) + \langle A_k x_k, \lambda \rangle + \frac{\rho}{2} \|x_k - x_k^t\|_{\Sigma_k}^2 \\ \text{s.t.} \quad & g_k(x_k) \leq 0, \quad h_k(x_k) = 0, \end{aligned} \tag{4}$$

where $\rho \geq 0$ is a penalty parameter, $\Sigma_k \succ 0$ is a weighting matrix, and λ is the dual variable. After solving (Eq. 4), the approximations of the constraint Jacobian $\tilde{\nabla} g_k(x_k^{(t+1)})$ and Hessian $H_k \approx \nabla^2_{x_k x_k}(f_k(x_k^t) + \gamma_k^\top g_k(x_k^{t+1}) + \mu_k^\top h_k(x_k^{t+1}))$ are computed, where γ_k and μ_k are the Lagrange multipliers; compared to ADMM, the use of more accurate Hessian and Jacobian approximations can reduce iterations at the cost of increased per-iteration complexity.

When exact second-order information is used, ALADIN can drive primal and dual residuals to machine precision, yielding solutions that are numerically indistinguishable from those of a centralized interior-point solver. With cheaper quasi-Newton updates (e.g. block-BFGS) the coordination QP is solved with an inexact curvature, yet empirical studies such as [42], [43] show objective gaps below 0.01% relative to the central baseline. Thus the cost of using approximate second-order information is merely a negligible loss in optimality—well inside normal dispatch tolerances—while reducing communication and computation per iteration.

Subsequently, a coupled coordination QP is solved at a central node:

$$\begin{aligned} \min_{\Delta x_k} \; & \sum_{k \in \mathcal{K}} \left(\frac{1}{2} \|\Delta x_k\|_{H_k}^2 + \langle \Delta x_k, \nabla f_k \rangle \right) + \lambda \|r\|_2 + \frac{\rho}{2} \|r\|_2^2, \\ \text{s.t.} \quad & \tilde{\nabla} g_k(x_k^t) \Delta x_k = 0, \quad r = \sum_{k \in \mathcal{K}} A_k (x_k^t + \Delta x_k) - b. \end{aligned}$$

Finally, the primal and dual variables are updated as $x_k^{t+1} = x_k^t + \Delta x_k$ and $\lambda^{t+1} = \lambda^t + \rho \left(\sum_{k \in \mathcal{K}} A_k x_k^{t+1} - b \right)$. Under mild assumptions, ALADIN converges to a local minimizer of the nonconvex problem from any feasible starting point when combined with the proposed globalization strategy [41]. Under suitable conditions, it achieves a quadratic or superlinear local convergence rate [41], matching centralized sequential QP methods.

ALADIN has been applied to AC OPF and power system analysis [43], [44], AC/DC hybrid systems [42], and heterogeneous power systems in both single-machine numerical simulations [44] and geographically distributed environments [45]. However, its increased per-step communication and scalability issues, particularly with inequality constraints, are drawbacks; [42] shows an improved ADMM outperforms ALADIN in scalability for the AC OPF problem. To address these, [43] employs approximation methods for H_k using blockwise and damped BFGS updates. Bi-level distributed ALADIN [46] eliminates the central coordinator in the coupled QP step by solving it with decentralized ADMM or conjugate gradient. Recent advancements include improved computing times for large-scale AC power flow problems using second-order corrections for linearization errors of active constraints in (Eq. 4) [47].

C. Distributed Interior Point Method

The Distributed Interior Point Method (IPM) is a promising approach for solving the large-scale nonconvex OPF problem. To overcome the limitations of extensive communication and central coordination of existing distributed second-order methods [48], distributed IPM reformulates (Eq. 1) by replacing inequality constraints with logarithmic barrier terms in the objective function:

$$\begin{aligned} \min_{\{x_k, \delta_k\}} \; & \sum_{k \in \mathcal{K}} \left(f_k(x_k) - \kappa \sum_{i=1}^{m_k} \ln(\delta_{k,i}) \right) \\ \text{s.t.} \quad & h_k(x_k) = 0, \; g_k(x_k) + \delta_k = 0, \; \delta_k \geq 0, \quad \forall k \in \mathcal{K}, \\ & \text{and Eq. 1b}, \end{aligned} \tag{5}$$

where $\kappa > 0$ is the barrier parameter, $\delta_k = \{\delta_{k,j}\}_{j=1,\ldots,m_k} \in \mathbb{R}^{m_k}$ are the slack variables. The main challenge in decentralization is solving the coupled linear system arising from the Karush-Kuhn-Tucker (KKT) conditions of (Eq. 5) in each Newton step. This difficulty is primarily due to the coupling constraints (Eq. 1b). To address this, methods have been developed to decentralize the computation of Newton steps. In [49], an incremental ADMM-based IPM is presented for distributed OPF with discrete variables. This approach consists of outer-loop iterations using an extended IPM that forms regional linear correction equations capturing the coupling relationships between neighboring areas. The inner-loop iterations employ ADMM to compute primal-dual directions in a distributed manner, allowing each region to solve its Newton step locally while coordinating with others through shared variables. Another strategy, proposed in [50], involves a two-stage optimization framework that decomposes the power network into a master network and subnetworks. By incorporating barrier terms into the subnetwork problems, the second-stage value function becomes differentiable with respect to the master problem variables. This smoothing facilitates the use of efficient nonlinear solvers based on primal-dual IPMs, which achieve *fast local convergence* because each outer iteration is a full Newton step on the KKT system (quadratic convergence near a KKT point), and they remain *communication-light in a distributed setting* because the power-grid's sparse topology results in a bordered block-diagonal KKT matrix. This structure allows the most intensive computation—matrix factorization—to be parallelized for each block (area), limiting communication to the small set of border variables that couple the system.

IV. Distributed Machine Learning Techniques

A. Deep Learning for Distributed OPF and Related Problems

Deep learning can accelerate distributed OPF computations, with neural networks emulating AC OPF solvers to achieve near-optimal solutions in milliseconds [53]. Many distributed OPF methods employ direct prediction (Fig. 3(b1)) to map grid conditions to control setpoints, known as solution functions (see [54] for Model Predictive Control (MPC) and DL connections). These approaches provide fast approximate solutions with minimal optimality loss and constraint violations [55], enforced via penalty terms [55] or primal-dual methods [56]. Decentralized schemes train local ML models to predict setpoints from local measurements [57]. Recent advancements [58] extend this by using DL to emulate Volt/VAR dynamics, providing scalable optimal design of incremental Volt/VAR control rules capturing Volt/VAR dynamics through recursive neural networks.

Incorporating uncertainty is vital due to renewable generation variability. This is addressed through chance constraints [59] or conditional value-at-risk (CVaR) [60], which provides a convex surrogate for chance constraints. Graph neural networks (GNNs) incorporate grid topology [61], with robustness against anomalous and missing measurements [62]. Attention networks have been employed alone [63] or with Convolutional Neural Networks (CNNs) [64]. Post-training, GNNs and CNNs enable distributed predictions using local computations based on limited neighboring information. Data-driven methods depend heavily on training data quality and coverage, with out-of-sample robustness and constraint satisfaction remaining key challenges due to frequent topology changes and DER uncertainty. Fig. 3 provides a visual overview of the spectrum of ML integration in optimization, ranging from enhancing existing algorithms (e.g., Learning-Assisted Optimization) to fully replacing them with ML models (e.g., Direct Prediction). The choice of approach depends on factors such as the desired level of computational speed, the need for generalization to new problem instances, and the specific characteristics of the power system application, as illustrated in Fig. 4.

1) Data Efficiency and Scalability: Sobolev training enhances data efficiency by incorporating sensitivities of the OPF solution function into the regression process [65]. Instead of the "OPF-then-learn" paradigm, decentralized policies can be directly integrated into the OPF problem ("OPF-and-learn"). For instance, [56] learns distributed nonlinear inverter controls via a deep neural network (DNN) with individualized inputs and partially connected layers, formulated under a chance-constrained framework and solved through gradient-free learning. To address scalability, [66] decomposes the power network into regions, first predicting coupling variables and then training region-specific models in parallel—scaling to large systems (up to 6700 buses) while maintaining feasibility and reducing training time.

While not yet widely explored in the context of distributed OPF, meta-learning approaches, as shown in Fig. 3(b4), hold promise for further enhancing data efficiency and adaptability by "learning to learn" across diverse tasks, enabling rapid adaptation with minimal data. Although direct applications in distributed OPF are limited, related works in other problems have demonstrated the effectiveness of meta-learning for improving generalization and sample efficiency in complex optimization problems. For instance, [67] introduces a voltage control strategy that adapts to faults using feature extraction and selective data filtering, while [68] proposes a meta-learning framework enabling rapid retraining for new OPF topologies. Similarly, [69] combines meta-strategy optimization with deep reinforcement learning (DRL) for emergency control, outperforming state-of-the-art DRL and model predictive control methods. Extending these advances to distributed OPF represents a promising direction for future research.

2) Constraint Satisfaction: Feasibility in distributed OPF is crucial for ensuring that operational decisions adhere to physical, regulatory, and safety constraints under dynamic conditions. Key constraints in this context typically involve maintaining voltage limits, adhering to power flow limits on transmission lines, ensuring generator output limits are not exceeded, and upholding system stability. The following outlines various approaches designed to address feasibility in power systems: 1) *Restricted feasible region during training*: Modifying the OPF feasible region to encourage models to produce strictly feasible solutions [70]. 2) *Active set prediction*: Predicting active constraints and solving a reduced DC OPF problem [71]. 3) *Physics-informed models*: Incorporating physical constraints into the loss function via penalty terms. In general, this approach is efficient at reducing, but not eliminating, feasibility violations and therefore have recently been combined with feasibility restoration methods (discussed next) [62], [64]. 4) *Feasibility restoration*: Post-processing or projection onto the feasible space using power flow solvers, e.g., predict-and-reconstruct in [55]. 5) *Implicit layers and gauge mappings*: Embedding feasibility restoration within the model using projection or gauge (one-to-one) mappings [72]. 6) *Control-theoretic safe synthesis*: Applying control-theoretic tools to define a feasible set for neural network weights that satisfy constraints [73]. Each approach has trade-offs. Methods 1 and 2 simplify learning but may struggle with complex feasible spaces or yield infeasible solutions. Methods 3 and 4 effectively reduce violations but lack strict guarantees. Bridging this gap, [74] propose a framework to establish worst-case guarantees for neural network predictions

TABLE I
COMPARISON OF DISTRIBUTED OPF METHODS

Method	Convergence (guarantees / rate)	Scalability	Communication per iteration	Computational cost per iteration
ADMM & Variants	Convex: global convergence, often $\mathcal{O}(1/k)$; nonconvex: local KKT point under assumptions.	Highly scalable: local subproblems solved in parallel; scales with size of local regions.	Low: exchange of primal/dual variables with neighbors or central hub.	Low to moderate: small local solves; many iterations may be needed.
ALADIN	Nonconvex: local KKT; locally quadratic rate near optimum; global convergence with line search [41].	Moderate: local NLPs parallelized, but central coordination QP size grows with coupling.	High: each agent sends sends primal solution *and* (approx.) Jacobian/Hessian; coordinator returns duals.	High: each iteration includes full NLP solves and a global QP; total iterations small.
Primal-Dual Interior-Point	Superlinear or quadratic local convergence; polynomial iteration complexity.	Good when exploiting sparsity; scales with number of tie-line buses [49], [50].	Moderate: communication per Newton step involves boundary variables and residuals.	Very high: Newton steps require solving large sparse systems; mitigated by parallelization.
ML-based methods (DL, RL, & MARL)	DL: Empirical only. Hard feasibility via safety layers/projection. MARL: Nash equilibrium in cooperative settings, though nonstationarity challenges convergence [51], [52].	High for DL with local models; moderate for distributed RL; MARL: combinatorial growth in joint action spaces but parameter sharing helps.	Low–moderate: DL requires minimal exchange post-training; distributed RL exchanges small gradients; MARL enables selective communication.	High training, very low inference: DL has expensive training but millisecond inference; RL moderate online learning; MARL moderate per-agent, scales with agents.

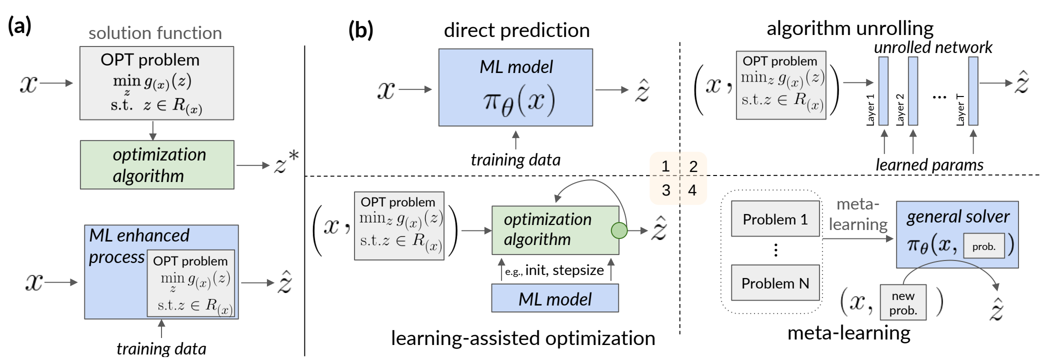

Fig. 3. **Integration of ML and optimization. (a)** Traditional optimization (direct problem solving) vs ML (learning from data). **(b)** Four ML-enhanced approaches: 1. **Direct Prediction:** ML maps parameters to optimal solutions; 2. **Algorithm Unrolling:** Converting iterative algorithms to trainable networks; 3. **Learning-Assisted Optimization:** ML enhances traditional algorithms for distributed control; 4. **Meta-Learning:** Training general strategies for rapid problem adaptation.

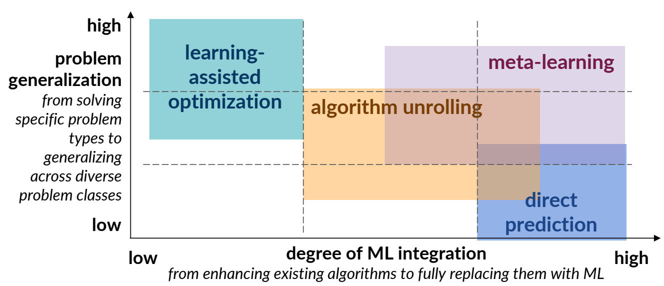

Fig. 4. Illustration of the four categories of ML-enhanced optimization methods, highlighting the trade-off between the degree of **ML integration** (from enhancing existing algorithms to fully replacing them with ML models) and **problem generalization** (adaptability to different problem types).

in OPF, providing a systematic method to ensure robust safeguards against constraint violations. Techniques 5 and 6 offer principled feasibility embedding but can be computationally expensive or rely on assumptions. Most methods are agnostic to DL architecture and can be used to learn distributed policies with sparse connections [56].

3) Perspective from Algorithm Unrolling: Deep learning architectures, such as DNNs, CNNs, GNNs, or recurrent neural networks (RNNs), can be viewed as the repeated application of an operator $\mathcal{F}^{(l)}$ across multiple layers $l \in \{1, ..., n_l\}$, i.e., $\mathcal{F}^{(n_l)} \circ \mathcal{F}^{(n_l-1)} \circ \cdots \circ \mathcal{F}^{(1)}$. This is reminiscent of iterative algorithms. For a simple illustration, the iterative algorithm $x^{(l+1)} = \sigma_{\rho/\kappa}(x^{(l)} + \frac{1}{\kappa}B^\top(c - Bx^{(l)}))$ can be used to solve $\min \frac{1}{2}\|Bx - c\|_2^2 + \rho\|x\|_1$, where $\sigma_{\rho/\kappa}$ is the element-wise soft-thresholding function $\sigma_\theta(x) = \text{sign}(x)\max(0, |x| - \theta)$ with $\theta = \rho/\kappa$ and κ usually taken as the largest eigenvalue of $B^\top B$. We can treat each iteration as an instantiation of the operator $\mathcal{F}_{ISTA}(x^{(l)}) = \sigma_{\theta^{(l)}}(W_1^{(l)}x^{(l-1)} + W_2^{(l)}c)$, where $(\theta^{(l)}, W_1^{(l)}, W_2^{(l)})$ are learnable parameters to train from data with $x^{(0)}$ as the initial point. This concept broadly connects to algorithm unrolling [75], where optimization algorithms are unrolled into DL architectures. By connecting to algorithm unrolling for algorithms such as ADMM, we can potentially develop distributed optimization algorithms that leverage the expressiveness and learning capabilities of DL models. Fig. 4 positions algorithm unrolling as ML-enhanced optimization with intermediate ML integration. It preserves optimization algorithm structure while learning parameters from data, balancing traditional method interpretability with DL flexibility.

This suits problems with specific structures requiring fast convergence.

B. Distributed RL for OPF and Related Problems

Distributed RL is well-suited for power system optimizations such as OPF, which involve high-dimensional spaces, complex constraints, and real-time decisions. In this context, distributed RL allows K agents —typically representing individual power generating units, storage devices, or load controllers—to operate independently or with varying degrees of coordination to make decisions based on local observations, aiming to optimize a combination of local and global objectives, such as minimizing cost, maximizing efficiency, or stabilizing the grid. At time step t, agent k observes state $s_t^k \in \mathcal{S}$, selects action $a_t^k \in \mathcal{A}$ according to policy π_{θ_k}, where θ_k represents the parameters of the policy, and receives reward $r_t^k = R(s_t^k, a_t^k)$. The environment transitions to state s_{t+1}^k according to $P(s_{t+1}^k | s_t^k, a_t^k)$. The goal is to find a set of policies $\{\pi_k\}_{k=1}^K$ that maximizes the expected cumulative discounted reward:

$$V(\{\pi_k\}_{k=1}^K) = \mathbb{E}_{\mathcal{T} \sim p(\mathcal{T} | \{\pi_k\}_{k=1}^K)} \left[\sum_{k=1}^{K} \sum_{t} \gamma^t r_t^k \right], \quad (6)$$

where $\mathcal{T} = \{s_t^k, a_t^k, r_t^k\}_{k,t}$ is the set of trajectory of states, actions, and rewards for all agents, and $p(\mathcal{T} | \theta_1, \ldots, \theta_K)$ is the probability distribution over trajectories induced by $\{\pi_{\theta_k}\}_{k=1}^K$.

Applying distributed RL to OPF requires consideration of the parallel interaction assumption, as agents can influence each other's states and rewards. While this can suit simulation or weakly decoupled systems, caution is needed to avoid potential divergence under distributional shifts [76]. Despite these challenges, insights from distributed RL, such as parallelization, stable learning, and simplified exploration, can inform the development of realistic RL-based solutions for power systems.

Enforcing OPF constraints in distributed RL. RL actions in OPF must satisfy hard nonlinear AC power flow constraints, making standard reward shaping or linear CMDP surrogates insufficient. Current research converges on three complementary strategies: (i) CMDP formulations with dual variables or penalties improve long-run compliance but allow instantaneous violations; (ii) *runtime assurance* filters modify RL actions in real-time via convex projection onto AC-feasible sets or control barrier functions providing safety certificates [77]; (iii) hierarchical designs where high-level RL proposes setpoints while lower-level controllers enforce exact AC constraints [78], [79]. Designing differentiable, back-propagatable safety layers for large multi-area grids remains a promising research direction.

C. Multi-Agent RL for Distributed OPF and Related Problems

MARL has shown promise for distributed optimization/control problems, where multiple agents coordinate actions in a shared environment [51], often modeled as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP).

In fully cooperative Dec-POMDPs, agents share a reward function and seek a joint policy $\pi = \{\pi_k\}_{k \in \mathcal{K}}$ that maximizes the expected discounted cumulative reward (Eq. 6) [51]. In contrast, competitive or mixed Dec-POMDPs involve agents with individual reward functions R_k aiming to maximize their own expected discounted return $V_k(\pi_k, \pi_{-k}) = \mathbb{E}_{\mathcal{T} \sim p(\mathcal{T} | \{\pi_k\}_{k=1}^K)} [\sum_t \gamma^t r_k^t]$ while considering others' policies π_{-k}. The resulting joint policy $\pi^* = \{\pi_k^*\}_{k=1}^K$ represents a Nash equilibrium, where $\pi_k^* \in \arg\max_{\pi_k} V_k(\pi_k, \pi_{-k}^*)$.

Most works assume fully cooperative agents [51], [52], [80]–[86], with a few considering coordination signal design [82], [87]. The non-cooperative setting in power system applications has been examined in online feedback equilibrium seeking [88]. Key MARL challenges include nonstationarity from concurrent updates, scalability from joint action space growth, and partial observability requiring efficient communication [89], [90]. Nonstationarity and scalability can hinder convergence to an optimal solution, while partial observability may lead to suboptimal local decisions misaligned with the collective goal.

1) Dealing with Nonstationarity: Centralized Training for Decentralized Execution (CTDE) allows agents to share information during training but act based on local observations during execution [52], [81], [82], [86]. MADDPG (Multi-Agent Deep Deterministic Policy Gradient) [91], a popular CTDE method, uses a centralized critic conditioned on all agents' observations and actions, while the actor only accesses local information. Although primarily applied to cooperative settings (e.g., [52]), MADDPG can also handle mixed cooperative-competitive environments. Off-policy learning enhances stability by learning from past experiences. Examples include MASAC (Multi-Agent Soft Actor-Critic) [83], off-policy maximum entropy RL [80], and Twin TD3 (Delayed Deep Deterministic Policy Gradient) [84], [92]. Maintaining a model of other agents, as in MADDPG or using techniques such as confederate image technology [86], is beneficial. [89] outlines five categories of handling nonstationarity. In power systems, the most common are ignoring (assuming stationarity) and forgetting (updating based on recent observations), with fewer works addressing response or opponent modeling.

2) Scalability: Parameter sharing, where all agents update a single set of network parameters, improves scalability by leveraging data from each agent and reducing policy oscillations [52], [92]. Integrating Graph Convolutional Networks (GCNs) further embeds topology information [52]. Exploration strategies such as parameter-space noise [85] and spatial discount factors [81] help contain the effective action space by focusing on local impacts. Open-source simulation platforms [51], [81] enable benchmarking of MARL methods, with MADDPG and TD3 exhibiting good scalability [51].

3) Handling Partial Observability: Local measurements are commonly used to achieve distributed optimization under partial observability [51], [52], [81], [83]. Recurrent networks, such as Gated Recurrent Units (GRUs) (e.g., [52]) and Long Short-Term Memory (LSTM) (e.g., [81]), can effectively encode history to extract relevant features. Learning a surrogate model using Sparse Variational Gaussian Processes (SVGP) to create a simulation environment for MARL [85] can reduce real-world communication and data collection. Agents modeling other agents [86], [91] can also mitigate partial observability.

4) Communication Efficiency: Communication allows agents to share information and coordinate actions, but it must be done efficiently. Some approaches assume no explicit communication (e.g., decentralized training), while selective schemes only exchange essential data, such as value/policy [80] or encoded states [81]. In structured communication, such as networked MARL (e.g., [80], [84]), each agent only needs to communicate with its neighbors. Agents can also learn communication protocols end-to-end, such as using differentiable communication [81]. To handle agent and communication failures, [80] proposes constructing replacement states using historical averages and local policy networks; [84] examine how communication topology changes affect learning.

Incentive mechanisms encourage collaboration among independent entities such as Distributed System Operators (DSOs) and DERs, aligning diverse goals through incentives rather than direct commands. Bi-level optimization integrates decisions across the power system hierarchy, bridging independent actions and collective goals. A cooperative bi-level framework [82] uses an asymmetric Markov game and bi-level actor-critic algorithm for real-time control. Similarly, [92] adopts a bi-level approach balancing operational safety and market interests. While existing approaches use penalty functions and global reward signals to promote cooperation and align objectives, [87] introduces the Markov Signaling Game for strategic incentive-compatible communication under information asymmetry, enabling efficient and stable policies.

D. Learning-Assisted Distributed Optimization Techniques

In contrast to Direct Prediction, Learning-Assisted Optimization techniques, as shown in Fig. 3(b3), aim to enhance existing optimization algorithms with ML, striking a balance between computational efficiency and generalizability. One powerful example of this approach is the integration of RL with various distributed optimization methods (such as those discussed in Sec. III) to tackle complex and stochastic nonlinear dynamic control problems. These include primal-dual decomposition and Lagrangian relaxation, where RL optimizes dual variables for faster convergence [79], [93]–[95]; Interior-Point Policy Optimization (IPPO) integrates RL with IPM for effective constraint handling [96]; and stochastic optimization incorporates RL to manage uncertainties [95]; and adaptive optimization [97], where RL is used to leverage and adapt the solution function of an optimization problem [54] as a policy function in a distributed setting.

Several studies have demonstrated the improved effectiveness of integrating ADMM with learning methods for solving OPF problems [98]–[100]. An asynchronous ADMM framework with momentum-extrapolation prediction has been introduced to manage asynchronous updates and communication failures [98]. RNNs have been applied to predict ADMM convergence rates, accelerating optimization while maintaining privacy [99]. ADMM's consensus parameter learning can be learned, optimizing decentralized power system efficiency [100]. Deep Q-learning has been employed to dynamically select optimal penalty parameters in ADMM for AC OPF, significantly reducing computational complexity [101]. While ADMM lends itself well to learning-based enhancements, other optimization methods from Sec. III, have also benefited from the integration of learning techniques. The flexibility of Learning-Assisted Optimization, highlighted in Fig. 4, allows for tailoring ML enhancements to specific components of existing optimization algorithms, potentially leading to improved performance without sacrificing interpretability. A comparison of key distributed OPF methods appears in Table I.

V. Distributed Real-Time Optimal Power Flow and Related Problems

RT-OPF leverages real-time grid data to address DER integration challenges [8]. It differs from standard OPF through: time-varying cost functions and constraints, rapid optimal solution tracking [102], and implicit determination of non-controllable variables by power flow equations. Applications range from sub-minute frequency regulation to hourly dispatch and daily storage scheduling [8].

While earlier works laid the groundwork for RT-OPF (see [8] for a review), recent distributed RT-OPF research leverages local measurements for improved robustness against single point of failure and plug-and-play integration of new grid components [103]–[107]. A case study on a 502-node distribution system demonstrated calculation time reduction to 2.34% of centralized methods [107].

For distributed optimization, the transition from static to real-time involves incorporating measurements (voltages, currents, and power flows) at the point of common coupling, exploiting physical laws for power flow solutions, and continuous information exchange. Primal-dual updates use these measurements for agent coordination and feasibility [103], [108] while correcting open-loop feedforward control inaccuracies through gradient computations. Combining precomputed linearized power flow models with real-time feedback manages nonlinearities without centralized Jacobians [105], [107], [109].

This methodology aligns with broader power system optimization practices, where algorithms serve as robust feedback controllers. Embedding optimization routines in physical system operations enhances grid adaptability [110]. These integrated control strategies enforce operational constraints and manage uncertainties with reduced model information and computational requirements, enabling resilient operations in rapidly changing environments.

A. Optimization Methods for Distributed RT-OPF

1) Distributed Formulation and Decomposition Methods: Real-time optimization extends static approaches by introducing time dependency in (Eq. 1). As in Sec. III, the main techniques include Lagrangian relaxation (LR) based decomposition, such as ADMM [103], dual ascent [105], [109], and regularized Lagrangian of the convex relaxation [104]. KKT-based decomposition, such as distributed interior point method [106], and primal decomposition with consensus constraints [111], [112] are also used.

Real-time measurements enable decoupling state sensitivities, allowing agents to predict states using local and neighbor information [107]. The changes in power flow states can be expressed as: $\Delta x_k(t+1) = S_{kk}(t) \times \Delta p_k(t) + \sum_{j\in\mathcal{N}_k^{\text{shared}}} S_{kj}(t) \times \Delta p_j(t)$, where $\Delta x_k(t+1)$ represents the predicted changes in power flow states, $S_{kk}(t)$ is the sensitivity submatrix, $\Delta p_k(t)$ represents the changes in DER output powers, and $\sum_{j\in\mathcal{N}_k^{\text{shared}}} S_{kj}(t) \times \Delta p_j(t)$ is the aggregated information from neighboring areas.

Another approach implicitly decomposes the OPF problem using learned local equilibrium functions $h_k^e(q_k, v_k)$ to map reactive power q_k and voltage v_k to optimal reactive power setpoints q_k^* [113]. This enables the decentralized control: $q_k(t+1) = q_k(t) + \epsilon(h_k^e(q_k(t), v_k(t)) - q_k(t))$, where $\epsilon \in [0,1]$ is the step size.

Hierarchical decomposition is also considered for coordination [108], [114]. A bi-level optimization in [115] uses upper level for DER group setpoints and lower level for individual DER disaggregation. Unlike spatial decomposition, [108] proposes temporal decomposition linking day-ahead and real-time markets through time-varying bi-level optimization. This approach requires clear connections between timescales, with distinct techniques for different horizons (e.g., distributionally robust optimization for day-ahead, online optimization for real-time operation).

2) Handling Time-Variation and Constraints: A common technique uses primal-dual gradient dynamics, viewing the path traced by variables satisfying KKT conditions as parameters vary [8]. The key challenge in distribution is decoupling subproblems while maintaining coordination: [106] uses first-order optimality conditions with boundary variables as parameters, where agents share quadratic approximations with a coordinator for optimal increment computation. Time-varying conditions are also handled through receding horizon MPC [111], [114], Lyapunov optimization [116], and online convex optimization [8], [117].

3) Information Exchange and Local Computation: Most distributed optimization requires a central coordinator (e.g., distribution management system [109], network operator [104], [115]). Agents send action information [104] and boundary variables [103], while coordinators broadcast dual variables (incentive signals) [104], [108] or primal variables (setpoints) [105] and monitor constraints [108]. Peer exchange involves boundary information like power flow states [107], [112] and local objective estimates for consensus [118].

Each agent's local computation often prefers simple methods such as projected gradient [118] or closed-form expressions [106] derived from KKT conditions. This is often achieved by leveraging appropriate linear approximations of the AC power-flow equations [103], [112], [119]. A fixed number of iterations of the distributed algorithm using the previous solution as a warm start may be performed [104]. In principle, ADMM with low-iteration complexity (Sec. III-A2) can be used [103] with transferable convergence analysis.

4) Other Practical Considerations: Unbalanced three-phase distribution systems utilize inter-phase coordination strategies [112]. Non-ideal communication effects, including delays and packet drops, show that moderate delays cause suboptimality but not instability [109], [112], [120]. Adaptive step size tuning accelerates convergence and avoids oscillations [109]. Network size, consensus steps, and gradient bias impacts on convergence have been assessed [118]. Local iterative updates using gradient estimates and projection optimize networked nonlinear system steady-state performance while circumventing local sensitivities and satisfying input constraints [118]. For real-time distributed equilibrium seeking, see [88].

B. Discussion of ML for Distributed RT-OPF

ML methods (Sec. IV) are inherently suitable for RT-OPF due to the fast response capability of the learned policies based on system states (e.g., [121]). MARL techniques (Sec. IV-C) rely on localized information and are inherently suitable for distributed counterparts. For instance, [122] demonstrates the real-time computational feasibility, with an online execution time of about 40ms for a 123-bus system.

Most papers apply offline-trained RL policies (e.g., RL [95] or safe RL methods [77]–[79], [123]) for online control without further adaptation. If the environment is nonstationary, the offline-trained policy may not perform optimally. Extensive pretraining with diverse conditions may help handle nonstationary environments [121], [123]. For instance, [121] considers uncertainties from renewable energy sources and $N-1$ topology changes during training, making the trained agent more robust during online implementation. Safety and stability are primary concerns in RT-OPF. Control-theoretic approaches can be used for stability-certified RL [73]. For safety constraints on state/action spaces, common approaches include penalty-based methods or using Lagrangian to derive primal-dual algorithms [78], [121], [124]. A knowledge-driven action masking technique is introduced to explicitly identify critical action dimensions based on the physical model, guiding the policy exploration in the safety direction [77]. A safe RL method based on Proximal-Dual Optimization-based Proximal Policy Optimization (PDO-PPO) algorithm is proposed [79], eliminating the need for manually selecting penalty weights between rewards and safety violations. A holomorphic embedding based safety layer in the RL policy can be added to ensure the operability of the control actions [124]. [123] introduces a supervisor-projector framework, wherein the supervisor evaluates RL-generated actions for safety, and the projector applies minimal modifications to ensure operational viability. In [78], a hybrid method enhances sample efficiency by deriving actor gradients through solving the KKT conditions of the Lagrangian using power system models.

For efficient utilization of computational resources and faster solution times, existing works e.g., [120], use techniques aligned with data parallelism in distributed ML. Despite the promise of RL for RT-OPF, most practical implementations rely on offline trained policies with linear constraint relaxations (e.g. CMDP or Lagrangian penalties). These simplifications cannot fully capture nonlinear AC power flow constraints, leading to conservative behavior or feasibility violations. Runtime assurance architectures, such as safety filters based on CBFs or AC power flow projections, can bridge this gap by filtering raw RL actions through certified nonlinear solvers while preserving learning-based adaptability [125].

VI. Key Challenges and Prospective Directions

Scalability and Computational Efficiency: Distributed approaches balance communication overhead, suboptimality, and fault susceptibility. ADMM, ALADIN, Analytical Target Cascading (ATC), and Auxiliary Problem Principle (APP) comparisons reveal wall-time versus iteration count discrepancies from local computation and communication overhead [126]. Practical benefits depend on communication infrastructure, data protocols, and local/central resource balance with accuracy and convergence. Joint consideration of communication and convergence is essential; data servers ensure accuracy while quantized messages reduce overhead. Hardware-aware design is crucial for low-resource settings.

Handling Nonstationarity, Uncertainty, and Stochasticity: Renewable integration, dynamic loads, and evolving topologies introduce nonstationarity and uncertainty in power systems. Distributed optimization dynamics add complexity through time-varying networks, asynchronous updates, and varying agent mechanisms (Sec. IV-C1). Online and real-time distributed methods with real-time data offer promising solutions (Sec. V). A promising direction is optimization algorithms with adaptive parameters responding to system and agent changes in topology, synchronization, and penalties. Incorporating power system domain knowledge into ML models can improve their data efficiency and generalization (Sec. IV-A), as demonstrated by the winning solution to the CityLearn Challenge [97]. Data-centric AI emphasizes data quality for robust ML models, which could be useful for managing the integrity of distributed data.

Drawing inspiration from "antifragility", optimization/ML methods can be designed to withstand and benefit from uncertainty and variability. Connections to meta-learning, continual learning, and multi-objective/quality-diversity optimization advance computational antifragility [127].

Privacy: Distributed optimization involving sensitive data exchange raises privacy concerns, as many existing methods, such as ALADIN, involve extensive data sharing with central coordinators, making systems vulnerable to honest-but-curious agents and external eavesdroppers [128]. Differential privacy offers efficient solutions, with algorithms co-designed to balance privacy mechanisms and coordination through optimized stepsizes, weakening factors, and noise distributions [128]. Quantized message exchange provides additional privacy while maintaining communication efficiency. Federated learning (FL) enables collaborative ML training on decentralized datasets without sharing privacy-sensitive data, while also reducing communication costs [129]. To avoid exposing important and possibly proprietary information, organizations (i.e., local nodes) typically impose tight security constraints on sharing modeling algorithms and data, which heavily limit collaborations. The paper [130] is the first to introduce the Assisted Learning (AL) framework, which is intended for local nodes to assist each other in supervised learning tasks without revealing any individual node's algorithm, data, or even task. While in FL a central controller orchestrates the learning and the optimization, in AL, the local nodes have protocols to assist each other's private learning tasks by iteratively exchanging nonsensitive information such as fitted residuals [130]. Such distributed learning techniques align well with the distributed structure of smart grids and can result in more robust and effective grid operations, warranting future research into secure and efficient aggregation protocols tailored for the hierarchical structure of grid operators and DER agents.

Safety, Robustness, and Cybersecurity: Distributed power algorithms must be secure and resilient against failures, with anomaly detection as one approach. Boundary defense mechanisms leveraging network sparsity to recover regions outside attacked areas [131] is promising yet underexplored. ML models should be safe and data-efficient, especially under distributional shift (Sec. IV). Adapting the AI model inspector framework [132], including stress-testing with adversarial examples and out-of-distribution generalization checks, is relevant. Just as power system equipment, ML models need periodic re-assessment and updates to maintain robustness, requiring ongoing monitoring throughout their lifecycle in distributed OPF systems.

Acknowledgment

Ahmad Al-Tawaha and Ming Jin are supported partially by NSF ECCS-2500368, NSF ECCS-2331775, Commonwealth Cyber Initiative, Amazon-Virginia Tech Initiative for Efficient and Robust Machine Learning. Javad Lavaei was supported by the U.S. Army Research Laboratory and the U.S. Army Research Office under Grant W911NF2010219, Office of Naval Research under Grant N000142412673, and NSF.

References

[1] D. K. Molzahn, F. Dörfler, H. Sandberg, S. H. Low, S. Chakrabarti, R. Baldick, and J. Lavaei, "A survey of distributed optimization and control algorithms for electric power systems," *IEEE Trans. on Smart Grid*, vol. 8, no. 6, pp. 2941–2962, 2017.

[2] T. Yang, X. Yi, J. Wu, Y. Yuan, D. Wu, Z. Meng, Y. Hong, H. Wang, Z. Lin, and K. H. Johansson, "A survey of distributed optimization," *Annual Reviews in Control*, vol. 47, pp. 278–305, 2019.

[3] N. Patari, V. Venkataramanan, A. Srivastava, D. K. Molzahn, N. Li, and A. Annaswamy, "Distributed optimization in distribution systems: Use cases, limitations, and research needs," *IEEE Trans. on Power Systems*, vol. 37, no. 5, pp. 3469–3481, 2021.

[4] T. Mühlpfordt, X. Dai, A. Engelmann, and V. Hagenmeyer, "Distributed power flow and distributed optimization—formulation, solution, and open source implementation," *Sustainable Energy, Grids and Networks*, vol. 26, p. 100471, 2021.

[5] H. Khaloie, M. Dolanyi, J.-F. Toubeau, and F. Vallée, "Review of machine learning techniques for optimal power flow," *Available at SSRN 4681955*, 2024.

[6] J. Adan, A. K. Srivastava *et al.*, "Distributed optimization for the DER-rich electric power grid," *Foundations and Trends® in Electric Energy Systems*, vol. 7, no. 1, pp. 1–62, 2023.

[7] F. Zohrizadeh, C. Josz, M. Jin, R. Madani, J. Lavaei, and S. Sojoudi, "A survey on conic relaxations of optimal power flow problem," *European Jrnl. of Operational Research*, vol. 287, no. 2, pp. 391–409, 2020.

[8] Z. Wang, W. Wei, J. Z. F. Pang, F. Liu, B. Yang, X. Guan, and S. Mei, "Online optimization in power systems with high penetration of renewable generation: Advances and prospects," *IEEE/CAA Jrnl. of Automatica Sinica*, vol. 10, no. 4, pp. 839–858, 2023.

[9] K. Sun and X. A. Sun, "A two-level ADMM algorithm for AC OPF with global convergence guarantees," *IEEE Trans. on Power Systems*, vol. 36, no. 6, pp. 5271–5281, 2021.

[10] S. Mhanna, A. C. Chapman, and G. Verbič, "Component-based dual decomposition methods for the OPF problem," *Sustainable Energy, Grids and Networks*, vol. 16, pp. 91–110, 2018.

[11] S. Mhanna, G. Verbič, and A. C. Chapman, "Adaptive ADMM for distributed ac optimal power flow," *IEEE Trans. on Power Systems*, vol. 34, no. 3, pp. 2025–2035, 2018.

[12] R. Madani, A. Kalbat, and J. Lavaei, "A low-complexity parallelizable numerical algorithm for sparse semidefinite programming," *IEEE Trans. on Control of Network Systems*, vol. 5, no. 4, pp. 1898–1909, 2017.

[13] S. Sojoudi and J. Lavaei, "Exactness of semidefinite relaxations for nonlinear optimization problems with underlying graph structure," *SIAM Journal on Optimization*, vol. 24, no. 4, pp. 1746–1778, 2014.

[14] Y. Yang, X. Guan, Q.-S. Jia, L. Yu, B. Xu, and C. J. Spanos, "A survey of ADMM variants for distributed optimization: Problems, algorithms and features," *arXiv preprint arXiv:2208.03700*, 2022.

[15] C. Chen, B. He, Y. Ye, and X. Yuan, "The direct extension of admm for multi-block convex minimization problems is not necessarily convergent," *Mathematical Programming*, vol. 155, no. 1, pp. 57–79, 2016.

[16] T.-H. Chang, M. Hong, and X. Wang, "Multi-agent distributed optimization via inexact consensus ADMM," *IEEE Trans. on Signal Processing*, vol. 63, no. 2, pp. 482–497, 2014.

[17] T. Baroche, P. Pinson, R. L. G. Latimier, and H. B. Ahmed, "Exogenous cost allocation in peer-to-peer electricity markets," *IEEE Trans. on Power Systems*, vol. 34, no. 4, pp. 2553–2564, 2019.

[18] Z. Wang, T. Ding, C. Mu, Y. Huang, M. Yang, Y. Yang, Z. Chen, Y. Lin, and M. Li, "An ADMM-based power system partitioned black-start and parallel restoration method considering high-penetrated renewable energy," *International Jrnl. of Electrical Power & Energy Systems*, vol. 155, p. 109532, 2024.

[19] A. Inaolaji, A. Savasci, S. Paudyal, and S. Kamalasadan, "Distributed optimal power flow in unbalanced distribution grids with non-ideal communication," *IEEE Trans. on Industry Applications*, 2023.

[20] W. Deng, M.-J. Lai, Z. Peng, and W. Yin, "Parallel multi-block ADMM with o (1/k) convergence," *Jrnl. of Scientific Computing*, vol. 71, pp. 712–736, 2017.

[21] B. He, L. Hou, and X. Yuan, "On full jacobian decomposition of the augmented lagrangian method for separable convex programming," *SIAM Journal on Optimization*, vol. 25, no. 4, pp. 2274–2312, 2015.

[22] Y. Shen, X. Zhang, and X. Zhang, "A partial PPA block-wise ADMM for multi-block linearly constrained separable convex optimization," *Optimization*, vol. 70, no. 3, pp. 631–657, 2021.

[23] P. Zhu, Z. Liu, K. Sun, L. Wang, P. Hu, N. Zhu, and D. Jiang, "Distributed model predictive control strategy based on block-wised alternating direction multiplier method for microgird clusters," *IET Generation, Transmission & Distribution*, vol. 16, no. 22, pp. 4630–4639, 2022.

[24] A. Buccini, P. Dell'Acqua, and M. Donatelli, "A general framework for ADMM acceleration," *Numerical Algorithms*, vol. 85, pp. 829–848, 2020.

[25] M. Aicardi and G. Ferro, "A newton-based dual update framework for ADMM in convex optimization problems," *IEEE Control Systems Letters*, 2024.

[26] G. França, D. P. Robinson, and R. Vidal, "A nonsmooth dynamical systems perspective on accelerated extensions of ADMM," *IEEE Trans. on Automatic Control*, vol. 68, no. 5, pp. 2966–2978, 2023.

[27] R. I. Boţ, E. R. Csetnek, and D.-K. Nguyen, "Fast augmented lagrangian method in the convex regime with convergence guarantees for the iterates," *Mathematical Programming*, vol. 200, no. 1, pp. 147–197, 2023.

[28] G. Hou and X. Jian, "Distributionally robust chance-constrained economic dispatch of multi-area electricity–gas–heat integrated energy systems," *Electric Power Systems Research*, vol. 217, p. 109090, 2023.

[29] X. Chen, J. Zhai, J. Shen, Q. Wang, S. Li, and S. Wang, "Distributed and distributionally robust chance-constrained scheduling for VSC-MTDC meshed AC/DC power networks," *IEEE Trans. on Network Science and Engineering*, 2024.

[30] X. Gao, B. Jiang, and S. Zhang, "On the information-adaptive variants of the ADMM: an iteration complexity perspective," *Jrnl. of Scientific Computing*, vol. 76, pp. 327–363, 2018.

[31] S. Lu, J. D. Lee, M. Razaviyayn, and M. Hong, "Linearized ADMM converges to second-order stationary points for non-convex problems," *IEEE Trans. on Signal Processing*, vol. 69, pp. 4859–4874, 2021.

[32] F. Huang, S. Chen, and H. Huang, "Faster stochastic alternating direction method of multipliers for nonconvex optimization," in *International conf. on Machine Learning*. PMLR, 2019, pp. 2839–2848.

[33] Y. Liu, F. Shang, H. Liu, L. Kong, L. Jiao, and Z. Lin, "Accelerated variance reduction stochastic ADMM for large-scale machine learning," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 43, no. 12, pp. 4242–4255, 2020.

[34] Y. Zeng, J. Bai, S. Wang, and Z. Wang, "A unified inexact stochastic ADMM for composite nonconvex and nonsmooth optimization," *arXiv preprint arXiv:2403.02015*, 2024.

[35] T.-H. Chang, M. Hong, W.-C. Liao, and X. Wang, "Asynchronous distributed ADMM for large-scale optimization—part i: Algorithm and convergence analysis," *IEEE Trans. on Signal Processing*, vol. 64, no. 12, pp. 3118–3130, 2016.

[36] W. Ananduta, C. Ocampo-Martinez, and A. Nedić, "A distributed augmented lagrangian method over stochastic networks for economic dispatch of large-scale energy systems," *IEEE Trans. on Sustainable Energy*, vol. 12, no. 4, pp. 1927–1934, 2021.

[37] Y. Su, S. Wu, Z. Wang, Z. Bao, and F. Liu, "Hierarchically distributed energy management in distribution systems: An error-tolerant and asynchronous approach," *IEEE Trans. on Smart Grid*, 2023.

[38] Z. Pan and M. Cannon, "Asynchronous ADMM via a data exchange server," *IEEE Trans. on Control of Network Systems*, 2024.

[39] Y. Zhou, X. Shi, L. Guo, G. Wen, and J. Cao, "A proximal ADMM-based distributed optimal energy management approach for smart grid with stochastic wind power," *IEEE Trans. on Circuits and Systems I: Regular Papers*, 2023.

[40] K. Sun and X. A. Sun, "Dual descent augmented lagrangian method and alternating direction method of multipliers," *SIAM Journal on Optimization*, vol. 34, no. 2, pp. 1679–1707, 2024.

[41] B. Houska, J. Frasch, and M. Diehl, "An augmented lagrangian based algorithm for distributed nonconvex optimization," *SIAM Journal on Optimization*, vol. 26, no. 2, pp. 1101–1127, 2016.

[42] N. Meyer-Huebner, M. Suriyah, and T. Leibfried, "Distributed optimal power flow in hybrid AC-DC grids," *IEEE Trans. on Power Systems*, vol. 34, no. 4, pp. 2937–2946, 2019.

[43] A. Engelmann, Y. Jiang, T. Mühlpfordt, B. Houska, and T. Faulwasser, "Toward distributed OPF using ALADIN," *IEEE Trans. on Power Systems*, vol. 34, no. 1, pp. 584–594, 2018.

[44] R. Bauer, X. Dai, and V. Hagenmeyer, "A shapley value-based distributed AC OPF approach for redispatch congestion cost allocation," in *Proceedings of the 14th ACM International conf. on Future Energy Systems*, 2023, pp. 109–113.

[45] X. Dai, A. Kocher, J. Kovačević, B. Dindar, Y. Jiang, C. N. Jones, H. Çakmak, and V. Hagenmeyer, "Ensuring data privacy in ac optimal power flow with a distributed co-simulation framework," *arXiv preprint arXiv:2402.01001*, 2024.

[46] A. Engelmann, Y. Jiang, B. Houska, and T. Faulwasser, "Decomposition of nonconvex optimization via bi-level distributed aladin," *IEEE Trans. on Control of Network Systems*, vol. 7, no. 4, pp. 1848–1858, 2020.

[47] X. Dai, J. Zhai, Y. Jiang, C. N. Jones, and V. Hagenmeyer, "Distributed ac optimal power flow for generic integrated transmission-distribution systems," *arXiv preprint arXiv:2308.13282*, 2023.

[48] A. Engelmann, Y. Jiang, T. Mühlpfordt, B. Houska, and T. Faulwasser, "Toward distributed OPF using ALADIN," *IEEE Trans. on Power Systems*, vol. 34, no. 1, pp. 584–594, 2019.

[49] W. Lu, M. Liu, S. Lin, and L. Li, "Incremental-oriented admm for distributed optimal power flow with discrete variables in distribution networks," *IEEE Trans. on Smart Grid*, vol. 10, no. 6, pp. 6320–6331, 2019.

[50] S. Tu, A. Wächter, and E. Wei, "A two-stage decomposition approach for ac optimal power flow," *IEEE Trans. on Power Systems*, vol. 36, no. 1, pp. 303–312, 2020.

[51] J. Wang, W. Xu, Y. Gu, W. Song, and T. C. Green, "Multi-agent reinforcement learning for active voltage control on power distribution networks," *Advances in Neural Information Processing Systems*, vol. 34, pp. 3271–3284, 2021.

[52] C. Mu, Z. Liu, J. Yan, H. Jia, and X. Zhang, "Graph multi-agent reinforcement learning for inverter-based active voltage control," *IEEE Trans. on Smart Grid*, 2023.

[53] K. Baker, "Emulating AC OPF solvers with neural networks," *IEEE Trans. on Power Systems*, vol. 37, no. 6, pp. 4950–4953, 2022.

[54] M. Jin, V. Khattar, H. Kaushik, B. Sel, and R. Jia, "On solution functions of optimization: Universal approximation and covering number bounds," in *Proceedings of the AAAI conf. on Artificial Intelligence*, vol. 37, no. 7, 2023, pp. 8123–8131.

[55] X. Pan, M. Chen, T. Zhao, and S. H. Low, "Deepopf: A feasibility-optimized deep neural network approach for ac optimal power flow problems," *IEEE Systems Journal*, vol. 17, no. 1, pp. 673–683, 2022.

[56] S. Gupta, V. Kekatos, and M. Jin, "Controlling smart inverters using proxies: A chance-constrained dnn-based approach," *IEEE Trans. on Smart Grid*, vol. 13, no. 2, pp. 1310–1321, 2021.

[57] R. Dobbe, O. Sondermeijer, D. Fridovich-Keil, D. Arnold, D. Callaway, and C. Tomlin, "Toward distributed energy services: Decentralizing optimal power flow with machine learning," *IEEE Trans. on Smart Grid*, vol. 11, no. 2, pp. 1296–1306, 2019.

[58] S. Gupta, V. Kekatos, and S. Chatzivasileiadis, "Optimal design of volt/var control rules of inverters using deep learning," *IEEE Trans. on Smart Grid*, 2024.

[59] S. Karagiannopoulos, P. Aristidou, and G. Hug, "Data-driven local control design for active distribution grids using off-line optimal power flow and machine learning techniques," *IEEE Trans. on Smart Grid*, vol. 10, no. 6, pp. 6461–6471, 2019.

[60] K. S. Ayyagari, R. Gonzalez, Y. Jin, M. Alamaniotis, S. Ahmed, and N. Gatsis, "Learning reactive power control polices in distribution networks using conditional value-at-risk and artificial neural networks," *Jrnl. of Modern Power Systems and Clean Energy*, vol. 11, no. 1, pp. 201–211, 2023.

[61] M. Gao, J. Yu, Z. Yang, and J. Zhao, "A physics-guided graph convolution neural network for optimal power flow," *IEEE Trans. on Power Systems*, 2023.

[62] D. Cao, J. Zhao, J. Hu, Y. Pei, Q. Huang, Z. Chen, and W. Hu, "Physics-informed graphical representation-enabled deep reinforcement learning for robust distribution system voltage control," *IEEE Trans. on Smart Grid*, 2023.

[63] C. Li, A. Kies, K. Zhou, M. Schlott, O. El Sayed, M. Bilousova, and H. Stöcker, "Optimal power flow in a highly renewable power system based on attention neural networks," *Applied Energy*, vol. 359, p. 122779, 2024.

[64] Q. Tran, J. Mitra, and N. Nguyen, "Learning model combining of convolutional deep neural network with a self-attention mechanism for ac optimal power flow," *Electric Power Systems Research*, vol. 231, p. 110327, 2024.

[65] M. Jalali, M. K. Singh, V. Kekatos, G. B. Giannakis, and C.-C. Liu, "Fast inverter control by learning the OPF mapping using sensitivity-informed gaussian processes," *IEEE Trans. on Smart Grid*, 2022.

[66] M. Chatzos, T. W. Mak, and P. Van Hentenryck, "Spatial network decomposition for fast and scalable AC-OPF learning," *IEEE Trans. on Power Systems*, vol. 37, no. 4, pp. 2601–2612, 2021.

[67] Y. Zhao, G. Zhang, W. Hu, Q. Huang, Z. Chen, and F. Blaabjerg, "Meta-learning based voltage control strategy for emergency faults of active distribution networks," *Applied Energy*, vol. 349, p. 121399, 2023.

[68] Y. Chen, S. Lakshminarayana, C. Maple, and H. V. Poor, "A meta-learning approach to the optimal power flow problem under topology reconfigurations," *IEEE Open Access Jrnl. of Power and Energy*, vol. 9, pp. 109–120, 2022.

[69] R. Huang, Y. Chen, T. Yin, Q. Huang, J. Tan, W. Yu, X. Li, A. Li, and Y. Du, "Learning and fast adaptation for grid emergency control via deep meta reinforcement learning," *IEEE Trans. on Power Systems*, vol. 37, no. 6, pp. 4168–4178, 2022.

[70] T. Zhao, X. Pan, M. Chen, and S. Low, "Ensuring DNN solution feasibility for optimization problems with linear constraints," in *The Eleventh International conf. on Learning Representations*, 2022.

[71] A. Stratigakos, S. Pineda, J. M. Morales, and G. Kariniotakis, "Interpretable machine learning for DC optimal power flow with feasibility guarantees," *IEEE Trans. on Power Systems*, 2023.

[72] L. Zhang, D. Tabas, and B. Zhang, "An efficient learning-based solver for two-stage DC optimal power flow with feasibility guarantees," *arXiv preprint arXiv:2304.01409*, 2023.

[73] F. Gu, H. Yin, L. El Ghaoui, M. Arcak, P. Seiler, and M. Jin, "Recurrent neural network controllers synthesis with stability guarantees for partially observed systems," in *Proceedings of the AAAI conf. on Artificial Intelligence*, vol. 36, no. 5, 2022, pp. 5385–5394.

[74] A. Venzke, G. Qu, S. Low, and S. Chatzivasileiadis, "Learning optimal power flow: Worst-case guarantees for neural networks," in *IEEE International conf. on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*. IEEE, 2020, pp. 1–7.

[75] V. Monga, Y. Li, and Y. C. Eldar, "Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing," *IEEE Signal Processing Magazine*, vol. 38, no. 2, pp. 18–44, 2021.

[76] T. Yu, A. Kumar, Y. Chebotar, K. Hausman, S. Levine, and C. Finn, "Conservative data sharing for multi-task offline reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 34, pp. 11 501–11 516, 2021.

[77] P. Wu, C. Chen, D. Lai, J. Zhong, and Z. Bie, "Real-time optimal power flow method via safe deep reinforcement learning based on primal-dual and prior knowledge guidance," *IEEE Trans. on Power Systems*, 2024.

[78] Z. Yan and Y. Xu, "A hybrid data-driven method for fast solution of security-constrained optimal power flow," *IEEE Trans. on Power Systems*, vol. 37, no. 6, pp. 4365–4374, 2022.

[79] B. Feng, J. Zhao, G. Huang, Y. Hu, H. Xu, C. Guo, and Z. Chen, "Safe deep reinforcement learning for real-time ac optimal power flow: A near-optimal solution," *CSEE Jrnl. of Power and Energy Systems*, 2024.

[80] Y. Gao, W. Wang, and N. Yu, "Consensus multi-agent reinforcement learning for volt-var control in power distribution networks," *IEEE Trans. on Smart Grid*, vol. 12, no. 4, pp. 3594–3604, 2021.

[81] D. Chen, K. Chen, Z. Li, T. Chu, R. Yao, F. Qiu, and K. Lin, "Powernet: Multi-agent deep reinforcement learning for scalable powergrid control," *IEEE Trans. on Power Systems*, vol. 37, no. 2, pp. 1007–1017, 2021.

[82] L. Hong, M. Wu, Y. Wang, M. Shahidehpour, Z. Chen, and Z. Yan, "MADRL-based DSO-customer coordinated bi-level volt/var optimization method for power distribution networks," *IEEE Trans. on Sustainable Energy*, 2024.

[83] Y. Chen, Y. Liu, H. Yin, Z. Tang, G. Qiu, and J. Liu, "Multiagent soft actor–critic learning for distributed ess enabled robust voltage regulation of active distribution grids," *IEEE Trans. on Industrial Informatics*, 2024.

[84] Z. Fan, W. Zhang, and W. Liu, "Multi-agent deep reinforcement learning based distributed optimal generation control of DC microgrids," *IEEE Trans. on Smart Grid*, 2023.

[85] S. Li, W. Hu, D. Cao, Z. Chen, Q. Huang, F. Blaabjerg, and K. Liao, "Physics-model-free heat-electricity energy management of multiple microgrids based on surrogate model-enabled multi-agent deep reinforcement learning," *Applied Energy*, vol. 346, p. 121359, 2023.

[86] S. Li, W. Hu, D. Cao, J. Hu, Q. Huang, Z. Chen, and F. Blaabjerg, "A novel MADRL with spatial-temporal pattern capturing ability for robust decentralized control of multiple microgrids under anomalous measurements," *IEEE Trans. on Sustainable Energy*, 2024.

[87] Y. Lin, W. Li, H. Zha, and B. Wang, "Information design in multi-agent reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[88] M. Ye, Q.-L. Han, L. Ding, and S. Xu, "Distributed Nash equilibrium seeking in games with partial decision information: A survey," *Proceedings of the IEEE*, vol. 111, no. 2, pp. 140–157, 2023.

[89] P. Hernandez-Leal, M. Kaisers, T. Baarslag, and E. M. De Cote, "A survey of learning in multiagent environments: Dealing with non-stationarity," *arXiv preprint arXiv:1707.09183*, 2017.

[90] K. Zhang, Z. Yang, and T. Başar, "Multi-agent reinforcement learning: A selective overview of theories and algorithms," *Handbook of reinforcement learning and control*, pp. 321–384, 2021.

[91] R. Lowe, Y. I. Wu, A. Tamar, J. Harb, O. Pieter Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," *Advances in neural information processing systems*, vol. 30, 2017.

[92] Z. Sun and T. Lu, "Collaborative operation optimization of distribution system and virtual power plants using multi-agent deep reinforcement learning with parameter-sharing mechanism," *IET Generation, Transmission & Distribution*, vol. 18, no. 1, pp. 39–49, 2024.

[93] Z. Yan and Y. Xu, "Real-time optimal power flow: A lagrangian based deep reinforcement learning approach," *IEEE Trans. on Power Systems*, vol. 35, no. 4, pp. 3270–3273, 2020.

[94] A. R. Sayed, C. Wang, H. I. Anis, and T. Bi, "Feasibility constrained online calculation for real-time optimal power flow: A convex constrained deep reinforcement learning approach," *IEEE Trans. on Power Systems*, vol. 38, no. 6, pp. 5215–5227, 2022.

[95] T. Wu, A. Scaglione, and D. Arnold, "Constrained reinforcement learning for predictive control in real-time stochastic dynamic optimal power flow," *IEEE Trans. on Power Systems*, 2023.

[96] Y. Ye, H. Wang, P. Chen, Y. Tang, and G. Strbac, "Safe deep reinforcement learning for microgrid energy management in distribution networks with leveraged spatial-temporal perception," *IEEE Trans. on Smart Grid*, 2023.

[97] V. Khattar and M. Jin, "Winning the citylearn challenge: adaptive optimization with evolutionary search under trajectory-based guidance," in *Proceedings of the AAAI conf. on Artificial Intelligence*, vol. 37, no. 12, 2023, pp. 14 286–14 294.

[98] A. Mohammadi and A. Kargarian, "Learning-aided asynchronous ADMM for optimal power flow," *IEEE Trans. on Power Systems*, vol. 37, no. 3, pp. 1671–1681, 2021.

[99] D. Biagioni, P. Graf, X. Zhang, A. S. Zamzam, K. Baker, and J. King, "Learning-accelerated ADMM for distributed DC optimal power flow," *IEEE Control Systems Letters*, vol. 6, pp. 1–6, 2020.

[100] T. W. Mak, M. Chatzos, M. Tanneau, and P. Van Hentenryck, "Learning regionally decentralized AC optimal power flows with ADMM," *IEEE Trans. on Smart Grid*, 2023.

[101] S. Zeng, A. Kody, Y. Kim, K. Kim, and D. K. Molzahn, "A reinforcement learning approach to parameter selection for distributed optimal power flow," *Electric Power Systems Research*, vol. 212, p. 108546, 2022.

[102] J. Mulvaney-Kemp, S. Fattahi, and J. Lavaei, "Smoothing property of load variation promotes finding global solutions of time-varying optimal power flow," *IEEE Trans. on Control of Network Systems*, vol. 8, no. 3, pp. 1552–1564, 2021.

[103] Y. Zhang, M. Hong, E. Dall'Anese, S. V. Dhople, and Z. Xu, "Distributed controllers seeking ac optimal power flow solutions using admm," *IEEE Trans. on Smart Grid*, vol. 9, no. 5, pp. 4525–4537, 2017.

[104] X. Zhou, E. Dall'Anese, L. Chen, and A. Simonetto, "An incentive-based online optimization framework for distribution grids," *IEEE Trans. on Automatic Control*, vol. 63, no. 7, pp. 2019–2031, 2017.

[105] I. Farhat, E. Ekomwenrenren, J. W. Simpson-Porco, E. Farantatos, M. Patel, and A. Haddadi, "A multi-area architecture for real-time feedback-based optimization of distribution grids," *arXiv preprint arXiv:2401.09694*, 2024.

[106] W. Lu, K. Xie, M. Liu, X. Wang, and L. Cheng, "Online decentralized tracking for nonlinear time-varying optimal power flow of coupled transmission–distribution grids," *IEEE Trans. on Power Systems*, 2023.

[107] X. Zhu, X. Hou, J. Li, G. Yan, C. Li, and D. Wang, "Distributed online prediction optimization algorithm for distributed energy resources considering the multi-periods optimal operation," *Applied Energy*, vol. 348, p. 121612, 2023.

[108] Y. Guo, X. Han, X. Zhou, and G. Hug, "Incorporate day-ahead robustness and real-time incentives for electricity market design," *Applied Energy*, vol. 332, p. 120484, 2023.

[109] J. Comden, J. Wang, and A. Bernstein, "Adaptive primal–dual control for distributed energy resource management," *Applied Energy*, vol. 351, p. 121883, 2023.

[110] A. Hauswirth, Z. He, S. Bolognani, G. Hug, and F. Dörfler, "Optimization algorithms as robust feedback controllers," *Annual Reviews in Control*, vol. 57, p. 100941, 2024.

[111] Y. Shi, H. D. Tuan, A. V. Savkin, C.-T. Lin, J. G. Zhu, and H. V. Poor, "Distributed model predictive control for joint coordination of demand response and optimal power flow with renewables in smart grid," *Applied Energy*, vol. 290, p. 116701, 2021.

[112] L. Wang, L. Xie, Y. Yang, Y. Zhang, K. Wang, and S.-j. Cheng, "Distributed online voltage control with fast PV power fluctuations and imperfect communication," *IEEE Trans. on Smart Grid*, 2023.

[113] Z. Yuan, G. Cavraro, and J. Cortés, "Constraints on OPF surrogates for learning stable local volt/var controllers," *IEEE Control Systems Letters*, 2023.

[114] Z. Zhang, F. F. da Silva, Y. Guo, C. L. Bak, and Z. Chen, "Double-layer stochastic model predictive voltage control in active distribution networks with high penetration of renewables," *Applied Energy*, vol. 302, p. 117530, 2021.

[115] A. Bernstein and E. Dall'Anese, "Real-time feedback-based optimization of distribution grids: A unified approach," *IEEE Trans. on Control of Network Systems*, vol. 6, no. 3, pp. 1197–1209, 2019.

[116] S. Fan, G. He, X. Zhou, and M. Cui, "Online optimization for networked distributed energy resources with time-coupling constraints," *IEEE Trans. on Smart Grid*, vol. 12, no. 1, pp. 251–267, 2020.

[117] T. Zhao, A. Parisio, and J. V. Milanović, "Distributed control of battery energy storage systems for improved frequency regulation," *IEEE Trans. on Power Systems*, vol. 35, no. 5, pp. 3729–3738, 2020.

[118] W. Wang, Z. He, G. Belgioioso, S. Bolognani, and F. Dörfler, "Online feedback optimization over networks: A distributed model-free approach," *arXiv preprint arXiv:2403.19834*, 2024.

[119] Y. Guo, X. Zhou, C. Zhao, L. Chen, and T. H. Summers, "Optimal power flow with state estimation in the loop for distribution networks," *IEEE Systems Journal*, 2023.

[120] H. Ipach, L. Fisser, C. Becker, and A. Timm-Giel, "Distributed utility-based real-time power flow optimization in ict-enabled low voltage distribution grids," *IET generation, transmission & distribution*, vol. 17, no. 13, pp. 2900–2925, 2023.

[121] Y. Zhou, W.-J. Lee, R. Diao, and D. Shi, "Deep reinforcement learning based real-time ac optimal power flow considering uncertainties," *Jrnl. of Modern Power Systems and Clean Energy*, vol. 10, no. 5, pp. 1098–1109, 2021.

[122] Y. Zhang, X. Wang, J. Wang, and Y. Zhang, "Deep reinforcement learning based volt-var optimization in smart distribution systems," *IEEE Trans. on Smart Grid*, vol. 12, no. 1, pp. 361–371, 2020.

[123] X. Yang, H. Liu, W. Wu, Q. Wang, P. Yu, J. Xing, and Y. Wang, "Reinforcement learning with enhanced safety for optimal dispatch of distributed energy resources in active distribution networks," *Jrnl. of Modern Power Systems and Clean Energy*, 2024.

[124] A. R. Sayed, X. Zhang, G. Wang, C. Wang, and J. Qiu, "Optimal operable power flow: Sample-efficient holomorphic embedding-based reinforcement learning," *IEEE Trans. on Power Systems*, 2023.

[125] K. L. Hobbs and et al., "Runtime assurance for safety-critical systems: An introduction to safety filtering approaches for complex control systems," *IEEE Control Systems Magazine*, vol. 43, no. 2, pp. 28–65, 2023.

[126] M. Alkhraijah, R. Harris, C. Coffrin, and D. K. Molzahn, "Powermodelsada: A framework for solving optimal power flow using distributed algorithms," *IEEE Trans. on Power Systems*, 2023.

[127] M. Jin, "Preparing for black swans: The antifragility imperative for machine learning," *arXiv preprint arXiv:2405.11397*, 2024.

[128] Y. Wang, "Privacy in multi-agent systems," *arXiv preprint arXiv:2403.02631*, 2024.

[129] W. Lim, N. Luong, D. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao., "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Communications Surveys and Tutorials*, vol. 22, no. 3, p. 2031–2063, 2020.

[130] X. Xian, X. Wang, J. Ding, and R. Ghanadan, "Assisted learning: A framework for multi-organization learning," *Advances in neural information processing systems*, vol. 33, pp. 14 580–14 591, 2020.

[131] M. Jin, J. Lavaei, S. Sojoudi, and R. Baldick, "Boundary defense against cyber threat for power system state estimation," *IEEE Trans. on Information Forensics and Security*, vol. 16, pp. 1752–1767, 2020.

[132] P.-Y. Chen and P. Das, "Ai maintenance: a robustness perspective," *Computer*, vol. 56, no. 2, pp. 48–56, 2023.