Linear-Complexity Unified Defense Against Deception Attacks in Distributed Economic Dispatch Using Cryptography and Machine Learning

Md. Mainul Islam, *Graduate Student Member, IEEE,* Abdulrahman Takiddin, *Member, IEEE,* Muhammad Ismail, *Senior Member, IEEE,* Hasan Kurban, and Erchin Serpedin, *Fellow, IEEE*

***Abstract*—Deception attacks, including false data injection, replay, and Byzantine manipulation, can corrupt broadcast price signals in distributed economic dispatch (DED), breaking fairness and increasing generation costs while remaining stealthy. Existing defenses are often attack-specific and rely on inter-supplier cross-verification, incurring $\mathcal{O}(n^2)$ communication overhead, and many provide probabilistic detection that can cause false alarms or missed attacks. To deterministically detect multiple deception attacks with $\mathcal{O}(n)$ complexity, this paper proposes a unified security layer for DED based on threshold Schnorr signatures, ensuring data integrity, freshness, and global consistency of price broadcasts. Upon detection, the framework triggers a machine-learning-based robust post-attack recovery mechanism that estimates the true marginal price from historical demand–price correlations, enabling suppliers to continue near-optimal updates. Supplier privacy is preserved via lightweight pairwise masking that reveals only aggregate supply without sacrificing accuracy. The proposed scheme tolerates up to $\lfloor (n-1)/3 \rfloor$ malicious suppliers among n. Experiments on the IEEE 14-bus system quantify efficiency loss under representative attacks. Large-scale studies on the IEEE 118-bus system demonstrate that Byzantine manipulation can increase the system cost by up to 47%. In contrast, the proposed scheme limits the resulting efficiency loss to 0.3% by leveraging an offline-trained LightGBM predictor that achieves an R^2 score of 0.985 on the test set.**

***Index Terms*—Deception attacks, distributed optimization, economic dispatch, elliptic curve cryptography, machine learning, power system security, Schnorr signature.**

I. Introduction

Cyberattacks on power companies have grown in both frequency and sophistication in recent years [1], [2]. As power grids evolve into highly interconnected digital systems, vulnerabilities in communication networks increase the risk to grid reliability [3]. Notable incidents such as the 2015 BlackEnergy and the 2016 CrashOverride attacks in Ukraine, as well as the 2023 cyberattack on Denmark's energy infrastructure, underscore these risks [1]. Large utilities face millions of malicious attempts each month, many of which remain undetected [4]. The average weekly attacks on utilities increased from 500 in 2020 to 1,100 in 2022 [3], with attacks on U.S. utilities rising by 70% in 2024 compared to 2023 [5]. While these attacks do not always target electricity pricing directly, they clearly highlight the growing cybersecurity challenges facing cyber-physical power systems.

A critical operation in power systems is economic dispatch (ED), which optimally allocates generator outputs to minimize total generation cost while satisfying system constraints. In large-scale and multi-region power systems, scalability and privacy concerns make centralized ED challenging. Distributed ED (DED) addresses this challenge by enabling parallel computation and balancing local decisions with global coordination using Lagrange multipliers (price signals) across a distributed network [15]–[17]. However, attackers can manipulate these multipliers by compromising communication channels to steer the optimization process toward biased but suboptimal dispatch, resulting in unfair profits and market inefficiencies while remaining stealthy [18]–[21]. Even small deviations can have significant economic impacts, making real-time electricity pricing highly susceptible to cyberattacks [22]. Since generators are re-dispatched every 5 to 15 minutes, persistent intrusions can accumulate over time, causing substantial financial losses to power providers [23]. Compared with disruption attacks that directly interrupt services by preventing convergence, deception attacks that manipulate optimization while preserving apparent convergence are much harder to detect.

Cryptographic techniques protect against deception attacks by ensuring message authenticity and integrity. While lightweight, symmetric-key methods pose significant security and privacy risks in distributed networks, since compromise of the shared key can expose the entire system. They

Md. Mainul Islam and Erchin Serpedin are with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843, USA (e-mail: mdmainul11@tamu.edu; eserpedin@tamu.edu). Abdulrahman Takiddin is with the Department of Electrical and Computer Engineering, FAMU-FSU College of Engineering, Florida State University, Tallahassee, FL 32310, USA (e-mail: a.takiddin@fsu.edu). Muhammad Ismail is with the Cybersecurity Education, Research, and Outreach Center (CEROC) and Department of Computer Science, Tennessee Tech University, Cookeville, TN 38505, USA (e-mail: mismail@tntech.edu). Hasan Kurban is with the College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar (e-mail: hkurban@hbku.edu.qa).

Corresponding author: Erchin Serpedin.

This research was supported by the National Science Foundation, USA, under Grants 2220347 and 2220346.

TABLE I
COMPARISON OF EXISTING DISTRIBUTED ECONOMIC DISPATCH SECURITY SCHEMES

Work	FDI	Replay	Byzantine	Approach	Privacy	Integrity	Consistency	Coordinator	Complexity
[6]	✓	✓	Partially	Blockchain consensus	✗	✓	✓	✓	$\mathcal{O}(n^2)$
[7]	✓	✓	✗	Weighted mean-subsequence reduced	✗	✗	✗	✗	$\mathcal{O}(n^2)$
[8]	✗	✓	✗	Neighbors monitoring, confidence-based isolation	✓	✗	✗	✗	$\mathcal{O}(n^2)$
[9]	✗	✓	✗	Neighbors monitoring, reputation-based isolation	✓	✗	✗	✗	$\mathcal{O}(n^2)$
[10]	✗	✗	✗	Homomorphic encryption (Paillier)	✓	✗	✗	✗	$\mathcal{O}(n^2)$
[11]	✗	✗	✗	State decomposition	✓	✗	✗	✗	$\mathcal{O}(n^2)$
[12]	✓	✗	✗	Logical/operational verification	✗	✗	✗	✓	$\mathcal{O}(n^2)$
[13]	✗	✗	✓	Trust-based weight allocation	✗	✗	✓	✗	$\mathcal{O}(n^2)$
[14]	✓	✗	✗	Logical operation-aided detection	✓	✗	✗	✗	–
Ours	✓	✓	✓	Cryptography for detection and ML for recovery	✓	✓	✓	✓	$\mathcal{O}(n)$

also lack non-repudiation [24], which is essential to prove message origin and agreement among multiple parties, and cannot support threshold signature constructions, which are required to provide fault tolerance when some participants act maliciously. Schnorr signatures over elliptic curve cryptography (ECC) address these limitations by allowing each party to use a unique private–public key pair and enabling consensus on messages via signature aggregation. Although the original Schnorr protocol [25] does not by itself provide fault tolerance, it can be extended to a threshold Schnorr scheme via Pedersen's distributed key generation (DKG), enabling fault-tolerant signature aggregation and ensuring data consistency even in the presence of a Byzantine coordinator and a bounded number of malicious suppliers.

Once an attack is detected, the affected supplier(s) must recover from the attack's effects. Machine learning (ML) can assist this recovery process by training on historical, attack-free dispatch data and using the trained model to estimate the true values of falsified price signals, thereby enhancing system resilience.

Preserving generator cost-function privacy is another key requirement in DED. Existing techniques such as differential privacy introduce noise and thus compromise optimization accuracy, whereas homomorphic encryption provides accuracy at the cost of substantial computational burden and network overhead [8], [9], [11]. These issues can be addressed by lightweight pairwise masking, whereby the aggregator or any eavesdropper can learn only the total supply without observing individual generator outputs, while avoiding the overhead of homomorphic encryption.

A. Related Work

Several studies have explored cyberattack mitigation in ED. Chen et al. [6] proposed a blockchain-based, security-constrained ED using a coordination committee and consensus to prevent manipulation of Lagrange multipliers. Wang et al. [7] developed a resilient DED using extremum filtering, though their attacks disrupt convergence rather than subtly increase costs, resembling disruption rather than deception. Li et al. [8] and Huang et al. [9] addressed replay attacks via redundant links, enabling full-node communication for detection. However, without majority-rule consensus, these schemes fail when nodes disagree on a unit's integrity.

Yan et al. [10] proposed a privacy-preserving ED with homomorphic encryption, later optimized by Chen [11] using state decomposition, but both lack data integrity assurance. Siu et al. [12] tackled false data injection (FDI) in centralized ED using multi-supplier verification, which detects inconsistencies but compromises privacy by requiring suppliers to reveal cost coefficients. Xing et al. [13] proposed trust-based weight allocation for Byzantine attacks, lacking explicit verification. Jena et al. [14] used event-driven FDI detection, but its sensitivity to minor fluctuations causes misclassification. Zhang et al. [26] demonstrated coordinated denial-of-service and FDI attacks to maximize costs while evading detection, but offered no mitigation.

As shown in Table I, most existing security studies on DED address only specific threats and incur quadratic communication complexity due to the elimination of a central coordinator and the resulting reliance on inter-supplier communication. While these decentralized approaches enhance resilience, they introduce scalability challenges and typically fail to guarantee both security and privacy. Ensuring security and privacy in DED with scalable solutions is therefore essential for maintaining market efficiency and grid reliability. This paper addresses this gap by combining threshold Schnorr signatures, pairwise masking, and machine learning (ML) into a unified framework. The proposed scheme covers three types of deception attacks such as FDI, replay, and Byzantine manipulation, which are more subtle than disruption attacks that prevent convergence.

B. Our Contributions

The primary contributions of this paper are as follows:

- We propose a unified security scheme for DED systems by modifying the original Schnorr protocol to enable verifiable consensus on price signals. Our approach safeguards against multiple deception attacks by providing integrity, freshness, consistency, and fault tolerance through deterministic detection mechanisms, in contrast to existing studies that tackle individual attacks separately and rely on weaker, probabilistic security.
- We ensure global data consistency with $\mathcal{O}(n)$ per-iteration communication complexity by eliminating inter-supplier interaction, thereby overcoming the $\mathcal{O}(n^2)$ overhead of existing DED security schemes and improving scalability for large electricity markets.

- While existing studies preserve generator cost-coefficient privacy only at the expense of optimality or computational efficiency, our scheme uses lightweight pairwise masking to protect cost-function privacy without degrading optimality and computational efficiency.
- We employ ML to reduce the negative impact of detected attacks by predicting the underlying price signals from historical demand–price correlations, thereby enhancing system resilience even when exact reconstruction is not possible.

II. Preliminaries

This section presents the DED formulation and the cryptographic tools used to protect against deception attacks.

A. Economic Dispatch

The ED problem optimizes power generation while satisfying system constraints. It is formulated as:

$$\min \sum_{i=1}^{n} C_i(P_i), \tag{1}$$

$$\text{subject to: } \sum_{i=1}^{n} P_i = D, \tag{2}$$

$$\underline{P}_i \le P_i \le \overline{P}_i, \quad \forall i \in \mathcal{N}. \tag{3}$$

Here, $C_i(P_i)$ represents the convex cost function of generator i, while P_i denotes its active power output (MW). The constraint in (2) ensures power balance, where D is the system demand. The set $\mathcal{N}$ contains all generators, and (3) enforces generator operational limits. For quadratic costs:

$$C_i(P_i) = a_i P_i^2 + b_i P_i + c_i, \quad \forall i \in \mathcal{N}, \tag{4}$$

where a_i, b_i, and c_i are the cost coefficients of generator i.

The Lagrangian function is given by:

$$L = \sum_{i=1}^{n} \left(a_i P_i^2 + b_i P_i + c_i\right) + \lambda \left(D - \sum_{i=1}^{n} P_i\right) + \sum_{i=1}^{n} \left[\underline{\mu}_i (P_i - \underline{P}_i) + \overline{\mu}_i (\overline{P}_i - P_i)\right], \tag{5}$$

where $\lambda \in \mathbb{R}$ represents the Lagrange multiplier corresponding to the power balance constraint (marginal price), while $\underline{\mu}_i$ and $\overline{\mu}_i$ denote the dual variables linked to the lower and upper generation limits of generator i, respectively.

The optimal generation satisfies the Karush-Kuhn-Tucker (KKT) conditions:

1) *Stationarity*:

$$2a_i P_i + b_i - \lambda + (\underline{\mu}_i - \overline{\mu}_i) = 0 \quad \forall i. \tag{6}$$

2) *Primal Feasibility*: Constraints (2) and (3) must hold.

3) *Dual Feasibility*:

$$\underline{\mu}_i, \overline{\mu}_i, \ge 0 \quad \forall i. \tag{7}$$

4) *Complementary Slackness*:

$$\underline{\mu}_i (P_i - \underline{P}_i) = 0, \quad \overline{\mu}_i (\overline{P}_i - P_i) = 0 \quad \forall i. \tag{8}$$

Algorithm 1 Distributed Economic Dispatch Overview

1: **Input:** Power grid
2: **Initialization:** The coordinator initializes $\lambda^{(0)}$, step size α, and sets $k = 0$.
3: **while** Not Converged **do**
4: Suppliers receive $\lambda^{(k)}$ from the coordinator.
5: Each supplier i computes and sends $P_i^{(k)}$ to the coordinator.
6: The coordinator aggregates $P_i^{(k)}$ and checks feasibility.
7: The coordinator updates $\lambda^{(k+1)}$.
8: The coordinator sends updated multipliers to suppliers.
9: $k \leftarrow k + 1$
10: **end while**
11: **Output:** Optimal generator outputs.

B. Distributed Economic Dispatch

DED methods, such as Lagrangian relaxation and the alternating direction method of multipliers (ADMM), decompose the centralized ED problem into subproblems and solve them in parallel [16], [17]. Each supplier determines its optimal generation based on received price signals. A coordinator maintains global feasibility and computes system-wide electricity prices. To clarify, the coordinator is a central entity responsible for the coordination of signals without exerting direct control over any supplier. This differs from centralized ED, where a central authority (i.e., the system operator) manages the entire power system by sending direct control commands to all suppliers [27].

Algorithm 1 outlines the DED procedure. The coordinator initializes the Lagrange multiplier $\lambda^{(0)}$ and transmits it to all suppliers. At iteration k, each supplier i computes its generator output using the KKT conditions:

$$P_i^{(k)} = \left[\frac{\lambda^{(k)} - b_i}{2a_i}\right]_{\underline{P}_i}^{\overline{P}_i}, \tag{9}$$

where the solution is projected onto the feasible interval $[\underline{P}_i, \overline{P}_i]$. Then, they send $P_i^{(k)}$ to the coordinator. The coordinator aggregates all $P_i^{(k)}$ and checks for the power balance. Convergence is declared if the change in λ is below a predefined threshold ε:

$$|\lambda^{(k)} - \lambda^{(k-1)}| < \varepsilon \text{ and } (D - \sum_{i=1}^{n} P_i^{(k)}) < \varepsilon. \tag{10}$$

If the conditions are not satisfied, λ is updated as:

$$\lambda^{(k+1)} = \lambda^{(k)} + \alpha \left(D - \sum_{i=1}^{n} P_i^{(k)}\right), \tag{11}$$

where α is a fixed step size that controls the convergence rate. The updated multiplier $\lambda^{(k+1)}$ is sent back to the suppliers to repeat the process until convergence is achieved.

C. Cryptographic Primitives

The proposed defense relies on three complementary cryptographic mechanisms:

- **Confidentiality:** Pairwise additive masking hides individual generator outputs while revealing only the aggregate supply.

- **Integrity and Freshness:** The elliptic curve digital signature algorithm (ECDSA) binds each message to a specific session and iteration, providing source authentication and replay protection.
- **Consistency, Efficiency, and Fault Tolerance:** Threshold Schnorr signatures instantiated via Pedersen distributed key generation (DKG) enable validation of a consistent price signal through a single group signature, while tolerating up to $T-1$ faulty signers in a committee of size n, where T is the signing threshold.

1) Elliptic Curve Cryptography: All public-key operations are instantiated using ECC, which operates on elliptic curves and provides equivalent security with shorter keys than RSA [24]. A widely adopted elliptic curve in practice is secp256k1, which is defined over the prime field $\mathbb{F}_p$ as [28]:

$$\mathcal{E} : y^2 = x^3 + 7 \bmod p, \tag{12}$$

where $p = 2^{256} - 2^{32} - 977$.

The curve $\mathcal{E}$ contains an additive, cyclic group of order q generated by a basepoint $B \in \mathcal{E}(\mathbb{F}_p)$. A public key Q is generated using elliptic curve scalar multiplication (ECSM) such that $Q = dB \in \mathcal{E}(\mathbb{F}_p)$ for a secret scalar $d \in \mathbb{Z}_q$.

ECDSA is used for message authentication since it is widely deployed as a digital signature scheme with well-studied security. It allows a user to generate a signature σ over a message m using their private key d, and verification is then performed using the corresponding public key Q. If valid, the signature guarantees message integrity and authenticates the signer, as only the holder of d could have produced it.

2) Schnorr Protocol: Over the secp256k1 group with basepoint B and group order q, the Schnorr protocol [25] is defined as follows.

Signing: Given a private key $\kappa \in \mathbb{Z}_q$ and message m, the signer:

1) Chooses a random nonce $r \in \mathbb{Z}_q$ and computes
$$R = rB. \tag{13}$$
2) Computes the challenge $\mathcal{C}$ using a hash function H:
$$\mathcal{C} = H(R \parallel m) \bmod q. \tag{14}$$
3) Computes the response
$$s = (r + \kappa\mathcal{C}) \bmod q. \tag{15}$$

The signature is the pair (R, s).

Verification: Given the public key $\mathcal{K} = \kappa B$, the verifier recomputes the challenge $\mathcal{C} = H(R \parallel m) \bmod q$ and checks

$$sB = R + \mathcal{C}\,\mathcal{K}. \tag{16}$$

The secp256k1 curve is selected because the linear relation $s = r + \kappa\mathcal{C}$ enables simple and efficient aggregation of partial signatures in the threshold setting.

3) Pedersen Distributed Key Generation: For robustness, the group signing key is generated distributively, without any trusted dealer, using Pedersen verifiable secret sharing (VSS) [29]. The protocol distributes a secret signing key across n parties so that any subset of at least T can perform threshold signing without reconstructing the key, while fewer than T learn nothing.

Each supplier i samples two random polynomials of degree at most $T-1$,

$$f_i(x) = \sum_{h=0}^{T-1} \beta_{i,h} x^h, \qquad g_i(x) = \sum_{h=0}^{T-1} \gamma_{i,h} x^h,$$

where the coefficients $\beta_{i,h}, \gamma_{i,h} \in \mathbb{Z}_q$ are chosen uniformly at random. The corresponding Pedersen commitments are

$$U_{i,h} = \beta_{i,h} B, \qquad V_{i,h} = \gamma_{i,h} B.$$

To party j, supplier i privately sends the scalar shares $f_i(j)$ and $g_i(j)$, which party j verifies through

$$\big(f_i(j) + g_i(j)\big)B = \sum_{h=0}^{T-1} j^h \big(U_{i,h} + V_{i,h}\big). \tag{17}$$

Shares that fail this check are discarded as invalid.

Each party j then aggregates all valid contributions into a single secret share

$$\kappa_j = \sum_{i=1}^{n} f_i(j) \bmod q, \qquad \mathcal{K}_j = \kappa_j B. \tag{18}$$

The group public key is $\mathcal{K} = \sum_{i=1}^{n} U_{i,0} = \kappa B$ for an implicit group secret $\kappa \in \mathbb{Z}_q$ that is never reconstructed. Pedersen DKG thus provides VSS with information-theoretic privacy and yields a group key that remains distributed.

4) Pairwise Masking: Individual generator outputs are kept confidential via additive pairwise masking in $\mathbb{Z}_M$ (typically $M = 2^{32}$), on top of the authenticated channel. For each edge (i, j) in a sparse masking graph, suppliers derive a shared secret using standard elliptic-curve Diffie–Hellman (ECDH) on secp256k1 [24]:

$$d_{i,j} = d_i Q_j = d_j Q_i, \tag{19}$$

where d_i and Q_i denote the private and public keys of supplier i. A pseudorandom function (PRF), instantiated as HMAC–SHA-256 keyed by $d_{i,j}$, expands this shared secret into a mask:

$$w_{i,j} = \mathrm{PRF}(d_{i,j}, \ell) \bmod M, \tag{20}$$

where ℓ is a domain-separation label. For the masking-neighbor set $\Omega(i)$, each supplier i computes

$$W_i = \sum_{j \in \Omega(i)} \mathrm{sgn}(i, j)\, w_{i,j}, \tag{21}$$

with $\mathrm{sgn}(i, j) \in \{+1, -1\}$ chosen so that $\sum_i W_i \equiv 0 \pmod M$. The transmitted quantity is

$$\widehat{m}_i = m_i + W_i \pmod M, \tag{22}$$

and the coordinator recovers only

$$\Big(\sum_i \widehat{m}_i\Big) \bmod M = \sum_i m_i \bmod M,$$

while each individual m_i remains hidden. Pairwise masking adds only lightweight operations, cancels exactly in the aggregate, and does not perturb the optimal ED solution.

III. Threat Model

This section formalizes deception attacks against distributed electricity markets along with privacy leakage that can strengthen these attacks. The threat model aligns with the mathematical formulation of DED and illustrates how attackers can reduce market efficiency by manipulating critical optimization parameters.

1) False Data Injection Attack:

Attacker Setup and Goal: In this insider attack, a malicious supplier exploits a man-in-the-middle position to alter Lagrange multipliers in transit. In the absence of cryptographic integrity protection, such manipulations are undetectable. The attacker's goal is to minimize a cheaper competitor's generator output and create an artificial supply shortage to maximize the malicious supplier's own profit while maintaining convergence and avoiding detection [22], [23], [30]–[33].

Attacker Strategy: The attacker persistently modifies the marginal price to $\tilde{\lambda}$, where $\tilde{\lambda} = \lambda - \tau$ and $\tau > 0$, before sending it to the targeted supplier.

Instead of receiving the correct marginal price λ, the victim receives the manipulated value $\tilde{\lambda}$, leading it to perceive reduced profitability and thus decrease its generation output according to (9).

As output from the cheaper supplier drops, an artificial supply shortage is introduced. This forces the system to rely more heavily on the malicious supplier, despite its higher costs, which in turn inflates the total generation cost and system marginal price. Meanwhile, the malicious supplier, operating with accurate multipliers, capitalizes on the inflated price at the expense of overall market efficiency. To remain stealthy, the attack is applied gradually.

2) Replay Attack:

Attacker Setup and Goal: In this outsider attack, the attacker intercepts and stores previously transmitted Lagrange multipliers without being able to modify or decrypt them. The attacker selectively replays outdated values to a subset of suppliers, especially those contributing significantly to the market. The goal is to increase generation cost through suboptimal dispatch, while maintaining convergence.

Attacker Strategy: At iteration k, the attacker targets high-impact suppliers and replays outdated multipliers $\lambda^{(k-j)}$ from iteration $k-j$; $0 < j < k$, while other suppliers receive the correct updated values. Thus, each supplier solves its local optimization problem using a potentially different view of the global state:

$$\lambda_i^{(k)} = \begin{cases} \lambda^{(k-j)}, & \text{if } i \in \mathcal{S}, \\ \lambda^{(k)}, & \text{otherwise}, \end{cases} \tag{23}$$

where $\mathcal{S}$ denotes the set of targeted suppliers.

This inconsistency results in suboptimal but convergent dispatch that elevates generation costs and avoids detection unless freshness verification mechanisms are in place.

3) Byzantine Attack:

Attacker Setup and Goal: In this stronger variant, the coordinator's private key is compromised, enabling the attacker to generate cryptographically valid signatures for manipulated messages. With this capability, the attacker sends different, yet cryptographically valid, marginal prices $\tilde{\lambda}_i$ to each supplier. The goal is to maximize generation cost while maintaining convergence and avoiding detection.

Optimal Attack Strategy: After observing the system for a few initial iterations, the attacker estimates the relative cost of each supplier based on their generation behavior, i.e., the ratio of generator output to maximum capacity. Suppliers with higher ratios are assumed to be cheaper, while those with lower ratios are considered more expensive.

Using this inferred cost ranking, the attacker customizes the manipulated price signal $\tilde{\lambda}_i$ for each supplier by applying a small symmetric perturbation τ to the true value λ:

- For the cheapest supplier: $\tilde{\lambda}_i = \lambda - \tau$, discouraging generation.
- For the most expensive supplier: $\tilde{\lambda}_i = \lambda + \tau$, encouraging maximum output.
- For mid-cost suppliers: $\tilde{\lambda}_i \in [\lambda - \tau,\ \lambda + \tau]$, maintaining power balance.

Each modified $\tilde{\lambda}_i$ is signed and sent to the corresponding supplier. Each supplier then solves its local optimization problem using a different marginal price, believing it to be legitimate due to the valid signature. This results in a globally inefficient dispatch that increases the overall generation cost. By gradually ramping up the manipulation over multiple iterations, the attack maintains stealth and evades detection mechanisms. However, the attack must persist across all iterations to succeed.

4) Eavesdropping or Privacy Leakage:

Attacker Setup and Goal: In this attack, an eavesdropper passively intercepts the communication between the coordinator and a targeted supplier [34], [35]. The primary goal of the attacker is to learn the supplier's cost coefficients, which are business-sensitive and confidential information.

Attacker Strategy: The attacker passively monitors the messages exchanged between the coordinator and supplier i during two consecutive iterations of the DED algorithm. In each iteration, the coordinator sends the updated Lagrange multiplier λ, and the supplier responds with its generator output P_i, computed using (9).

By recording two successive pairs, $(P_i^{(1)}, \lambda^{(1)})$ and $(P_i^{(2)}, \lambda^{(2)})$, the attacker obtains two linear equations:

$$\begin{cases} 2a_i P_i^{(1)} + b_i = \lambda^{(1)} \\ 2a_i P_i^{(2)} + b_i = \lambda^{(2)} \end{cases}$$

Solving this system yields the cost coefficients a_i and b_i:

$$a_i = \frac{\lambda^{(2)} - \lambda^{(1)}}{2\big(P_i^{(2)} - P_i^{(1)}\big)}, \quad b_i = \lambda^{(1)} - 2a_i P_i^{(1)}$$

This allows the attacker to extract the supplier's private cost parameters, violating confidentiality and potentially enabling market manipulation or targeted disruptions.

Mitigation Challenges: Each attack type requires a specific mitigation strategy. Replay attacks can be mitigated with timestamps and sequence numbers. Privacy and FDI attacks can be mitigated by encrypting and signing messages to ensure data confidentiality and integrity, but attackers can still exploit inconsistencies, leading to Byzantine faults, where an attacker deliberately sends conflicting yet plausible information to different parties. Mitigating Byzantine faults typically requires a consensus mechanism. The key challenge is ensuring all suppliers receive identical, unaltered multipliers from the coordinator. One approach is peer-to-peer (P2P) cross-verification among suppliers, but this requires $n(n-1)$ extra messages per iteration, increasing communication overhead. Therefore, a scalable security scheme is essential to comprehensively address these challenges.

IV. Proposed Unified Security Scheme

We integrate a unified security layer with Algorithm 1 so that every supplier receives one identical, authenticated, and fresh price signal $\lambda^{(k)}$ per iteration. The design eliminates P2P messaging, preserves privacy via pairwise masking, and provides succinct, fault-tolerant finality via threshold Schnorr signatures. It is assumed that at least T suppliers in the DED network are honest, with $\lfloor 2n/3 \rfloor < T \le n$.

Setup: Suppliers first execute the Pedersen DKG protocol (without involving the coordinator) over the secp256k1 group, obtaining additive shares $\kappa_i \in \mathbb{Z}_q$ and public shares $\mathcal{K}_i = \kappa_i B$, together with a group public key $\mathcal{K}$ for an implicit group secret $\kappa \in \mathbb{Z}_q$ that is never reconstructed. A threshold signing committee $\mathcal{T} \subseteq \{1, \dots, n\}$ of size $|\mathcal{T}| = T$ is fixed, and for each $i \in \mathcal{T}$ the Lagrange coefficient at zero is precomputed as

$$\omega_i(0; \mathcal{T}) = \prod_{\substack{j \in \mathcal{T} \\ j \neq i}} \frac{-j}{i-j} \bmod q. \tag{24}$$

Each supplier i and the coordinator hold an ECDSA keypair on secp256k1, denoted (d_i, Q_i) and (d_c, Q_c), for authenticating messages. The coordinator is assumed to know all suppliers' public keys $\{Q_i\}_{i=1}^n$ and to distribute Q_c to all suppliers. A sparse undirected masking graph is fixed so that each supplier i has

$$|\Omega(i)| = \left\lfloor \frac{n-1}{3} \right\rfloor + 1$$

neighbors, ensuring at least one honest neighbor under up to $\lfloor \frac{n-1}{3} \rfloor$ colluding adversaries. For each edge (i, j), parties establish an ECDH secret as $d_{i,j} = d_i Q_j = d_j Q_i$, which is expanded by a PRF into per-edge masks as in Section II-C4.

All parties agree on the threshold T, the convergence tolerance ε, a DED session identifier $\mathcal{I}$, an upper bound on iterations $K_{\max}$, and the initial Lagrange multiplier $\lambda^{(0)}$. For each edge (i, j) and each iteration k with $1 \le k \le K_{\max}$, a label $\ell(\mathcal{I}, k, i, j)$ is defined for use with the PRF to derive per-iteration masks; these masks (and the resulting zero-sum terms $W_i^{(k)}$) are precomputed for all $k \le K_{\max}$ to reduce online computation.

A. Proposed Optimization Process

Fig. 1 illustrates the operational workflow of the proposed scheme. Each iteration is organized into four communication phases, resulting in at most $4n$ messages per iteration.

1) **Local Optimization and Masking:** At iteration k, each supplier $i \in \mathcal{N}$ receives $\lambda^{(k)}$ and computes its local generator output according to (9), obtaining $P_i^{(k)}$. It then retrieves the precomputed zero-sum mask term $W_i^{(k)}$ and forms the masked output

$$\widehat{P}_i^{(k)} = P_i^{(k)} + W_i^{(k)} \pmod M. \tag{25}$$

Next, it selects a one-time random nonce $r_i^{(k)} \in \mathbb{Z}_q$ and computes the Schnorr nonce commitment $R_i^{(k)} = r_i^{(k)} B$. The supplier prepares the uplink message

$$m_i^{(k)} = \widehat{P}_i^{(k)} \parallel \mathcal{I} \parallel k \parallel R_i^{(k)} \tag{26}$$

and signs it using ECDSA:

$$\sigma_i^{(k)} = \mathrm{ECDSA.sign}\big(d_i, m_i^{(k)}\big). \tag{27}$$

It then transmits the signed tuple $\big(m_i^{(k)}, \sigma_i^{(k)}\big)$ to the coordinator.

2) **Output Aggregation and Dual Update:** The coordinator verifies each $\sigma_i^{(k)}$ using Q_i and checks that the counter $(\mathcal{I}, k)$ is current. Upon successful validation, it recovers the aggregate masked sum (pairwise masks cancel in the aggregate):

$$P^{(k)} = \Big(\sum_{i=1}^n \widehat{P}_i^{(k)} \Big) \bmod M = \sum_{i=1}^n P_i^{(k)}. \tag{28}$$

Decoding via the fixed-point representation yields $\sum_i P_i^{(k)}$ without revealing any individual $P_i^{(k)}$.
The coordinator then updates the Lagrange multiplier using (11). In parallel, it aggregates the Schnorr nonce commitments over the signing committee $\mathcal{T}$ as $R^{(k)} = \sum_{i \in \mathcal{T}} R_i^{(k)}$. It then prepares the coordinator message

$$m_c^{(k)} = \lambda^{(k+1)} \parallel R^{(k)} \parallel \mathcal{I} \parallel k \tag{29}$$

and signs it with ECDSA:

$$\sigma_c^{(k)} = \mathrm{ECDSA.sign}\big(d_c, m_c^{(k)}\big). \tag{30}$$

The pair $\big(m_c^{(k)}, \sigma_c^{(k)}\big)$ is broadcast to all suppliers, which verify $\sigma_c^{(k)}$ under Q_c and derive the common Schnorr challenge $\mathcal{C}^{(k)} = H\big(m_c^{(k)}\big) \bmod q$.

3) **Partial Signature Generation:** Each supplier $i \in \mathcal{T}$ uses its secret share κ_i, nonce $r_i^{(k)}$, and Lagrange coefficient $\omega_i(0; \mathcal{T})$ to form a partial Schnorr response on the common challenge $\mathcal{C}^{(k)}$:

$$s_i^{(k)} = r_i^{(k)} + \mathcal{C}^{(k)}\, \omega_i(0; \mathcal{T})\, \kappa_i \bmod q. \tag{31}$$

Each signer sends $s_i^{(k)}$ to the coordinator.

4) **Signature Aggregation:** For each received $s_i^{(k)}$, the coordinator uses the public share $\mathcal{K}_i = \kappa_i B$ and the

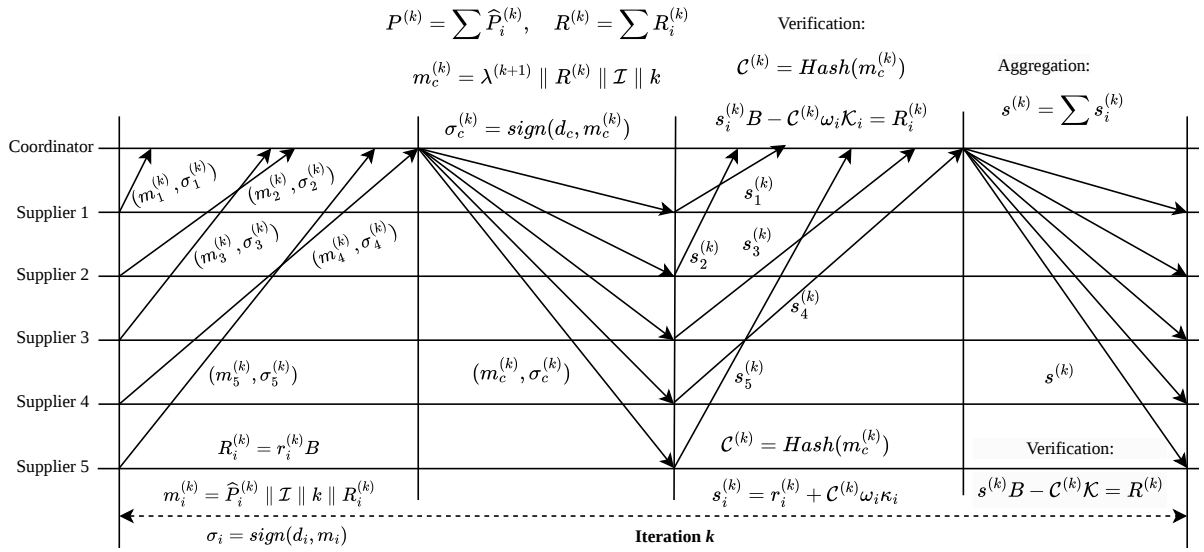

Fig. 1. Operational workflow in the proposed scalable security scheme for DED.

stored $R_i^{(k)}$ to verify the partial signature via

$$s_i^{(k)} B = R_i^{(k)} + \mathcal{C}^{(k)}\, \omega_i(0; \mathcal{T})\, \mathcal{K}_i. \tag{32}$$

Invalid partials are discarded. If at least T valid partials are available, the coordinator aggregates them into the threshold Schnorr signature by computing

$$s^{(k)} = \sum_{i \in \mathcal{T}} s_i^{(k)} \bmod q, \tag{33}$$

which, together with $R^{(k)}$ (already contained in $m_c^{(k)}$), forms a standard Schnorr signature on $m_c^{(k)}$ under the group public key $\mathcal{K}$. The coordinator then broadcasts $\left(m_c^{(k)}, s^{(k)}\right)$ to all suppliers. Each supplier verifies

$$s^{(k)} B = R^{(k)} + \mathcal{C}^{(k)}\, \mathcal{K}, \tag{34}$$

which ensures that all honest suppliers accept the same price update $\lambda^{(k+1)}$ or jointly reject it if the threshold signature check fails.

B. Extension to Distributed Unit Commitment

Although this work focuses on DED, the proposed security scheme is also applicable to distributed unit commitment (UC) problems, which determine the optimal scheduling of generating units by jointly deciding their on/off status $u_{i,t}$ and dispatch levels $P_{i,t}$ across a time horizon. Since the cryptographic layer is message-agnostic and operates on the serialized optimization payload, extending from single-period ED to multi-period UC requires no structural change in the security mechanism.

Consider a time horizon $t = 1, 2, \ldots, 24$ (i.e., 24 hours) with time-varying demand $\{D_t\}_{t=1}^{24}$. In a distributed UC iteration k, the coordinator broadcasts the time-dependent price vector $\boldsymbol{\lambda}^{(k)} = [\lambda_1^{(k)}, \ldots, \lambda_{24}^{(k)}]^\top$, and each supplier i solves its local UC problem to obtain commitment and dispatch trajectories $\mathbf{u}_i^{(k)} = [u_{i,1}^{(k)}, \ldots, u_{i,24}^{(k)}]$, $u_{i,t}^{(k)} \in \{0, 1\}$ and $\mathbf{P}_i^{(k)} = [P_{i,1}^{(k)}, \ldots, P_{i,24}^{(k)}]$, respectively. The privacy layer applies component-wise masks across time,

$$\widehat{P}_{i,t}^{(k)} = P_{i,t}^{(k)} + W_{i,t}^{(k)} \pmod M, \ \sum_{i=1}^{n} W_{i,t}^{(k)} = 0, \ \forall t. \tag{35}$$

The coordinator recovers the aggregate supply trajectory $P_t^{(k)} = \sum_{i=1}^n \widehat{P}_{i,t}^{(k)} \equiv \sum_{i=1}^n P_{i,t}^{(k)}$ and performs the regular multi-period dual update

$$\lambda_t^{(k+1)} = \lambda_t^{(k)} + \alpha\big(D_t - P_t^{(k)}\big), \qquad \forall t. \tag{36}$$

The coordinator broadcast message is

$$m_{c,\mathrm{UC}}^{(k)} = \mathrm{enc}\Big(\boldsymbol{\lambda}^{(k+1)}\Big) \parallel R^{(k)} \parallel \mathcal{I} \parallel k, \tag{37}$$

and the aggregate verification retains the same form,

$$s^{(k)} B = R^{(k)} + \mathcal{C}_{\mathrm{UC}}^{(k)}\, \mathcal{K}, \tag{38}$$

where the challenge $\mathcal{C}_{\mathrm{UC}}^{(k)}$ is the hash of the received coordinator message, such that $\mathcal{C}_{\mathrm{UC}}^{(k)} = H\Big(m_{c,\mathrm{UC}}^{(k)}\Big) \bmod q$.

Hashing the encoding of a scalar λ in ED and hashing the encoding of a vector $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_{24}]$ in UC are identical operations under deterministic serialization, and $\boldsymbol{\lambda}$ is not reconstructed from the hash. The coordinator transmits the actual values while the hash-based challenge binds the signature to those values to ensure integrity and global consistency. Consequently, preventing equivocation, replay, or forgery of $\boldsymbol{\lambda}^{(k+1)}$ directly protects UC decisions derived locally from the authenticated and threshold-consistent price trajectory across the entire horizon.

C. ML-Based Post-Attack Recovery

A mismatch in the session/iteration counter indicates a replay attack, failure of ECDSA verification indicates FDI, and failure of the aggregate threshold-Schnorr verification

indicates Byzantine behavior. Any such event flags manipulation of the communicated price signal.

Upon detection, an affected supplier i falls back to an ML-based estimate of the market-clearing price, trained offline from historical operating data. Let $\mathbf{x}_D^{\text{his}} \in \mathbb{R}^{F_D}$ and $\mathbf{x}_\lambda^{\text{his}} \in \mathbb{R}^{F_\lambda}$ denote the historical demand- and price-derived feature vectors, respectively. For the current demand D, the marginal price is estimated as $\lambda_i^{\text{rec}} = f_{\text{ML}}\big(D, \mathbf{x}_D^{\text{his}}, \mathbf{x}_\lambda^{\text{his}}\big)$, where $f_{\text{ML}}(\cdot)$ denotes the trained regression model.

Using the predicted price, supplier i updates its generator output via the local ED KKT mapping:

$$P_i^{\text{rec}} = \left[\frac{\lambda_i^{\text{rec}} - b_i}{2a_i}\right]_{\underline{P}_i}^{\overline{P}_i}. \tag{39}$$

Although λ_i^{rec} may not exactly match the true marginal price, it provides an attack-resilient approximation that steers P_i^{rec} toward a near-optimal dispatch and limits welfare loss until secure coordination is restored.

V. Security Analysis

We analyze the security of the proposed DED scheme with the following assumptions:

- The ECDSA and Schnorr schemes are existentially unforgeable under chosen-message attacks (EUF-CMA)-secure under the hardness of the elliptic curve discrete logarithm problem (ECDLP) [24].
- Communication channels between suppliers and the coordinator are authenticated but not private.
- Adversaries attempt to manipulate the dispatch results but avoid preventing convergence, which would yield no solution. They cannot break cryptographic primitives under security parameter η, the bit-length of private keys that bounds any probabilistic polynomial-time (PPT) adversary's success probability to $\mathsf{negl}(\eta)$.

Theorem 1 (Data Integrity and Freshness)**:** *Assume that: (a) all parties enforce counter checks in every iteration; (b) ECDSA and Schnorr signatures are* EUF-CMA*-secure under the hardness of the ECDLP; and (c) the signed content explicitly includes $(\mathcal{I}, k)$. Then, for any PPT adversary $\mathcal{A}$, the probability that an honest party accepts a staggered replay as a fresh, valid message for the current session and iteration is* $\mathsf{negl}(\eta)$.

Proof. Each message m and challenge $\mathcal{C}$ include the current session identifier $\mathcal{I}$ and iteration number k, which are bound to the signatures $\sigma^{(k)}$ and $s^{(k)}$. A staggered replay of an older iteration or session fails because the received pair $(\mathcal{I}, k)$ does not match the locally expected values, and the honest party deterministically discards the message.

If a PPT adversary $\mathcal{A}$ modifies the counter of an outdated packet to the current value while keeping the other content unchanged, the resulting message is a new, previously unseen string. Making this modified message acceptable requires producing a valid signature on an unseen message, which corresponds to an EUF-CMA forgery. By assumption, ECDSA and Schnorr signatures are EUF-CMA-secure under the computational hardness of ECDLP, hence

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{replay}}(\eta) \;\le\; \mathsf{Adv}_{\mathcal{A}}^{\text{ECDLP}}(\eta) \;=\; \mathsf{negl}(\eta).$$

□

Theorem 2 (Data Consistency and Fault Tolerance)**:** *Under the assumption that the ECDSA and Schnorr schemes are* EUF-CMA*-secure, the proposed DED framework maintains data consistency against any PPT adversary controlling the coordinator and up to $n - T$ suppliers, provided that in every iteration k at least $\lfloor 2n/3 \rfloor + 1$ honest suppliers are operational and the threshold satisfies $T > \lfloor 2n/3 \rfloor$.*

Proof. Let $\mathcal{H}^{(k)}$ and $\mathcal{M}^{(k)}$ denote the honest and malicious suppliers at iteration k, with $|\mathcal{H}^{(k)}| \ge T$ and $|\mathcal{M}^{(k)}| \le n - T$. In each iteration k, every honest supplier $i \in \mathcal{H}^{(k)}$ issues at most one partial Schnorr signature on the unique coordinator message $m_c^{(k)}$, while malicious suppliers may *double–sign* (sign multiple messages for the same k).

Consistency under honest coordinator: If the coordinator is honest but an adversary $\mathcal{A}$ modifies $m_c^{(k)}$ to $m'^{(k)}$ in transit, it corresponds to an EUF-CMA forgery, which contradicts the assumption. Since honest suppliers reject tampered messages, $\mathcal{A}$ is unable to impersonate the coordinator.

Consistency under Byzantine coordinator: Consider a biased coordinator sending inconsistent messages $m_c[i]^{(k)} \neq m_c[j]^{(k)}$ to honest suppliers i and j. If $\mathcal{T}^{(k)}$ contains indices that signed different messages $m \neq m'$, then $s^{(k)} = \sum_{i \in \mathcal{T}} s_i^{(k)} \bmod q$ cannot satisfy $s^{(k)}B = R^{(k)} + H(x)\,\mathcal{K}$, for any single x unless $H(m) = H(m')$, which would imply a collision or a signature forgery. Thus, each valid aggregate necessarily consists of partials on a *single* message.

Resilience against collusion attacks and necessity of supermajority: Assume for contradiction that $T \le 2n/3$. The coordinator partitions honest suppliers into $\mathcal{H}_1, \mathcal{H}_2$ with:

$$|\mathcal{H}_1| = \lfloor |\mathcal{H}^{(k)}|/2 \rfloor, \quad |\mathcal{H}_2| = \lceil |\mathcal{H}^{(k)}|/2 \rceil$$

and sends m to $\mathcal{H}_1$, $m' \neq m$ to $\mathcal{H}_2$. Malicious suppliers collude with the coordinator and sign both messages. Then:

$$|\mathcal{M}^{(k)}| + |\mathcal{H}_1| \ge (n - T) + \lfloor T/2 \rfloor \ge T$$
$$|\mathcal{M}^{(k)}| + |\mathcal{H}_2| \ge (n - T) + \lceil T/2 \rceil \ge T.$$

Both inequalities hold when $T \le 2n/3$. Thus the coordinator can produce valid signatures for both m and m', breaking consistency. To preclude such *collusion* for all $|\mathcal{M}^{(k)}| \le n - T$, it is *necessary and tight* that $T > \lfloor 2n/3 \rfloor$. □

Theorem 3 (Collusion Resistance)**:** *Let $\kappa \in \mathbb{Z}_q$ be generated via a Pedersen DKG yielding a T-of-n Shamir sharing. Assume the Schnorr protocol is* EUF-CMA*-secure. For any PPT adversary $\mathcal{A}$ corrupting a set $\mathcal{M} \subseteq [n]$,*

$$\Pr\big[\mathcal{A} \textit{ reconstructs } \kappa \vee \textit{ forges } s\big] \le \begin{cases} \mathsf{negl}(\eta) + \frac{1}{q}, & |\mathcal{M}| < T, \\ 1, & |\mathcal{M}| \ge T, \end{cases}$$

where $1/q$ is the guessing bound for κ.

Proof. Secret reconstruction: Pedersen DKG realizes a Shamir sharing with $(T-1)$-privacy: any view consisting of fewer than T valid shares is statistically independent of κ. Hence, for $|\mathcal{M}| < T$, the adversary's advantage in learning κ is *zero*; its best strategy is guessing, which succeeds with probability at most $1/q$. If $|\mathcal{M}| \geq T$, Lagrange interpolation of T shares recovers κ with probability 1.

Aggregate forgery: Fix any corrupted set $\mathcal{M}$ with $|\mathcal{M}| < T$. Suppose that $\mathcal{A}$ outputs a pair (m^*, s^*) that is accepted by an honest supplier without $\mathcal{A}$ having obtained T valid honest partial signatures corresponding to $H(m^*)$. We construct a forger $\mathcal{F}$ against the EUF-CMA security of the Schnorr signature scheme. The forger $\mathcal{F}$ receives a verification key $\mathcal{K}$ from its challenger and simulates the protocol, including the DKG transcripts, so that the aggregate public key equals $\mathcal{K}$. It answers all honest-signing queries by forwarding the corresponding canonical coordinator messages to its signing oracle. If $\mathcal{A}$ succeeds, then either (i) m^* was never queried to the oracle (direct forgery) or (ii) the oracle was queried on some $m \neq m^*$ (substitution forgery). In both cases $\mathcal{F}$ breaks EUF-CMA and thus $\Pr[\mathcal{A} \text{ forges } s] \leq \mathsf{negl}(\eta)$.

Combining: For $|\mathcal{M}| < T$, the union bound gives

$$\Pr[\mathcal{A} \text{ reconstructs } \kappa \vee \text{forges } s] \;\leq\; \underbrace{1/q}_{\text{guessing}} \;+\; \underbrace{\mathsf{negl}(\eta)}_{\text{EUF-CMA}}.$$

Under the system requirement $T > \lfloor 2n/3 \rfloor$, any adversary restricted to $|\mathcal{M}| \leq n - T < T$ cannot reconstruct κ and can only attempt forgeries, which are $\mathsf{negl}(\eta)$ by EUF-CMA.

If $q \geq 2^\eta$, then $1/q$ is negligible, yielding an overall negligible bound. For $|\mathcal{M}| \geq T$, reconstruction succeeds with probability 1, as stated.

□

Theorem 4 (Generator Cost Coefficient Privacy)**:** *Assume the PRF is secure under multi-instance usage with domain separation and the ECDH is secure. If the adversary corrupts at most $\lfloor \frac{n-1}{3} \rfloor$ suppliers (which may include the coordinator), then for any PPT adversary $\mathcal{A}$ and any honest supplier i, the sequence of masked values $\{\widehat{P}_i^{(k)}\}_{k=1}^K$ reveals no information about the cost coefficients (a_i, b_i) beyond what can be inferred from the aggregate system behavior.*

Proof. Let $\mathcal{M}$ be the set of corrupted suppliers with $|\mathcal{M}| \leq \lfloor \frac{n-1}{3} \rfloor$. This set may include the coordinator. For any honest supplier i, consider the mask $W_i^{(k)} = \sum_{j \in \Omega(i)} \mathrm{sgn}(i,j) \cdot w_{i,j}^{(k)}$. Due to the graph connectivity condition $|\Omega(i)| \geq \lfloor \frac{n-1}{3} \rfloor + 1$ and the corruption bound $|\mathcal{M}| \leq \lfloor \frac{n-1}{3} \rfloor$, each honest supplier i has at least one honest neighbor $j^* \in \Omega(i)$.

The domain separation in $\ell(\mathcal{I}, k, i, j)$ ensures that:

- Each $w_{i,j}^{(k)}$ is computed from a unique PRF input.
- All masks are computationally independent across iterations and supplier pairs.
- The PRF security holds even when the adversary sees multiple related outputs.

The mask $w_{i,j^*}^{(k)}$ for the honest edge (i, j^*) remains hidden from $\mathcal{A}$ and is computationally indistinguishable from random by the combined security of ECDH and the domain-separated PRF. Even knowing all other terms in $W_i^{(k)}$, the unknown $w_{i,j^*}^{(k)}$ randomizes the sum, making the transmitted parameter $\widehat{P}_i^{(k)} = P_i^{(k)} + W_i^{(k)}$ indistinguishable from random. Any adversary distinguishing $\widehat{P}_i^{(k)}$ from random with non-negligible advantage would break either the PRF or ECDH security, contradicting our assumptions.

Since the cost coefficients (a_i, b_i) influence optimization only through $P_i^{(k)}$, which is perfectly hidden by the random masks, the coordinator and any eavesdropper cannot learn (a_i, b_i) from $\widehat{P}_i^{(k)}$. If $|\mathcal{M}| \geq \lfloor \frac{n-1}{3} \rfloor$, privacy of some honest suppliers who could have all neighbors corrupted would be compromised due to exposing $W_i^{(k)}$. □

VI. Case Studies and Results

A. Threat Model Validation on IEEE 14-Bus

To validate the threat models and quantify their impact on market efficiency, we implemented the DED procedure in Algorithm 1 on the IEEE 14-bus system. The test system contains five generators owned by five independent suppliers, with capacity limits G_1 (0–332.4 MW), G_2 (0–140 MW), and G_3–G_5 (0–100 MW). Each supplier follows a quadratic production cost $C_i(P_i) = a_i P_i^2 + b_i P_i + c_i$, with coefficients $\{a_i, b_i\} = \{0.01, 32\}$, $\{0.05, 31\}$, $\{0.04, 33\}$, $\{0.01, 34\}$, and $\{0.02, 30\}$ for G_1–G_5, respectively. The parameters are arranged to induce a clear cost hierarchy in which Supplier 5 is the cheapest, Supplier 4 is the second cheapest, and Supplier 3 is the most expensive, enabling systematic evaluation of targeted manipulation.

Table II reports dispatch outcomes under normal operation and under deception attacks. In the baseline, low-cost suppliers contribute most of the demand, yielding a system marginal price of \$34.19/MWh and a total cost of \$8456.25/h. In the FDI attack, a malicious insider (e.g., Supplier 1) tampers with the price signal received by the cheapest competitor (Supplier 5), injecting an artificially low multiplier. The victim then reduces output to zero, forcing dispatch onto higher-cost units and increasing total cost to \$8730.83/h. In the replay attack, stale multipliers are replayed to Suppliers 1 and 5 while others receive fresh values. The resulting inconsistency produces a suboptimal yet convergent dispatch and increases cost to \$8600.35/h. The combined FDI+replay case amplifies these effects by simultaneously suppressing low-cost supply and preventing timely adaptation of another influential supplier.

We evaluate two Byzantine-capability regimes. In the Byzantine case, the adversary sends supplier-specific multipliers to systematically suppress low-cost generation while encouraging high-cost output. The attack effectiveness depends on how accurately the adversary can rank suppliers by cost. If ranking is inferred only from observed supply levels, the adversary may misclassify critical units; if cost coefficients are learned from intercepted traffic, the adversary can rank units correctly and maximize loss. The Privacy+Byzantine case models a stronger adversary that first

TABLE II
IEEE 14-BUS DISPATCH OUTCOMES UNDER NORMAL OPERATION AND DECEPTION ATTACKS.

Condition	Targets	Iter.	G_1		G_2		G_3		G_4		G_5		Price ($/MWh)	Total Cost ($/h)
			Output (MW)	Profit ($/h)	Output (MW)	Profit ($/h)	Output (MW)	Profit ($/h)	Output (MW)	Profit ($/h)	Output (MW)	Profit ($/h)		
Normal	–	21	109.25	119.36	14.81	8.78	3.08	0.29	31.85	50.72	100.00	218.50	34.19	8456.25
Replay	G_1, G_5	58	**49.75**	191.45	41.82	69.96	39.10	45.85	53.46	142.88	**74.87**	363.01	36.35	8600.35
FDI	G_5	25	165.33	273.33	28.83	33.25	21.78	14.23	43.07	92.73	**0.00**	0.00	35.31	8730.83
Byzantine	G_1, G_3, G_5	65	**0.00**	0.00	76.38	233.36	**100.00**	211.05	81.11	328.90	**1.51**	13.74	39.11	9342.53
FDI + Replay	G_5; G_1	61	**49.75**	286.56	65.72	172.75	70.96	151.05	72.57	263.35	**0.00**	0.00	38.26	9034.93
Privacy + Byzantine	All; G_1, G_2, G_5	38	**1.31**	7.87	**120.33**	25.64	67.10	135.08	70.26	246.83	**0.00**	0.00	38.03	9433.33

infers private cost coefficients via eavesdropping, enabling a more accurate ranking and a more damaging, systematic equivocation strategy. Consistent with the threat model, the Byzantine variants produce the largest welfare loss because they can steer many suppliers simultaneously.

B. ML-Guided Recovery on IEEE 118-Bus

Dataset Construction: To evaluate post-attack recovery at scale, we implemented DED on the IEEE 118-bus system with 54 generators. This system provides a standardized, meshed-network topology historically derived from the American Electric Power (AEP) system in the U.S. Midwest, enabling reproducible resilience evaluations at a moderate scale. As a realistic demand reference, we collected hourly system load time series for 2024–2025 from the official PJM website[1]. Although PJM's service territory differs from the historical AEP footprint underlying the IEEE 118-bus case, we use PJM load trajectories solely for their realistic temporal variability. To adapt the PJM data to the test system, we scaled the hourly load profile to match the total load level of the IEEE 118-bus case, preserving realistic demand dynamics while maintaining feasibility for the network constraints.

To obtain physically consistent "ground-truth" clearing prices for ML supervision, we solved ED for each hourly load using time-varying generator cost functions. In particular, we grouped generators into four power-plant types (coal, gas, nuclear, and wind) according to their default cost coefficients in the IEEE 118-bus case. Units with the largest coefficients are treated as gas-fired, whereas units with the smallest coefficients are treated as wind. Historical U.S. commodity price time series for coal, natural gas, and uranium were collected from Business Insider commodity market pages[2] and normalized to $[0, 1]$ via min–max scaling as shown in Fig. 2. Wind units are modeled with negligible fuel-cost variability. Using these normalized trajectories, we construct time-varying *linear* cost coefficients consistent with each unit's assigned fuel type while keeping the *quadratic* terms fixed to reflect unit efficiency, and then solve ED to obtain an hourly clearing-price sequence. From

[1]https://dataminer2.pjm.com/feed/hrl_load_metered/
[2]https://markets.businessinsider.com/commodities/

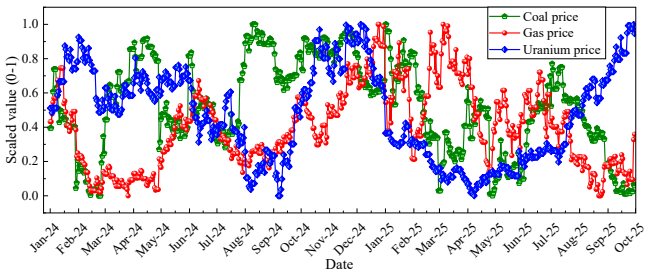

Fig. 2. Historical fuel price trajectories normalized to $[0, 1]$.

the resulting demand and price time series, we form the historical features $\mathbf{x}_D^{\text{his}}$ and $\mathbf{x}_\lambda^{\text{his}}$ used by the ML predictor.

Training and Testing: The ML model inputs consist of the current demand and historical demand- and price-derived features. The training dataset spans January–December 2024 and contains 8,783 hourly samples. The testing dataset covers January–September 2025 with 6,551 samples. Figures 3 and 4 illustrate the demand–price time series and their correlations for the training and testing datasets, respectively, confirming strong nonlinear dependence. Multiple regression models were evaluated, including SVR, Random Forest, Gradient Boosting, XGBoost, and LightGBM. Table III summarizes the testing performance. LightGBM achieves the highest R^2 (generalization) score of 0.9851 and therefore is adopted for post-attack price recovery.

Attack Scenarios and Recovery Results: We evaluated replay, FDI, Byzantine, and combined attacks on the IEEE 118-bus DED implementation, both without security and with the proposed cryptographic layer and ML fallback. The most dominant units under normal dispatch are G30, G40, G5, and G37. Accordingly, the attacker targets (i) G30 and G40 for individual replay and FDI, and (ii) a split set (FDI on G30/G40 and replay on G5/G37) for the combined case. For Byzantine attacks without cost inference, the attacker uses output-based heuristics to select dominant and expensive units; with privacy compromise, the attacker can infer costs and perform cost-aware targeting more effectively.

Table IV shows that, without security, all attacks increase the marginal price and total cost and typically require more iterations to converge. With the proposed scheme, manipulation is detected deterministically via integrity, freshness, and consistency failures, and affected suppliers fall back to the

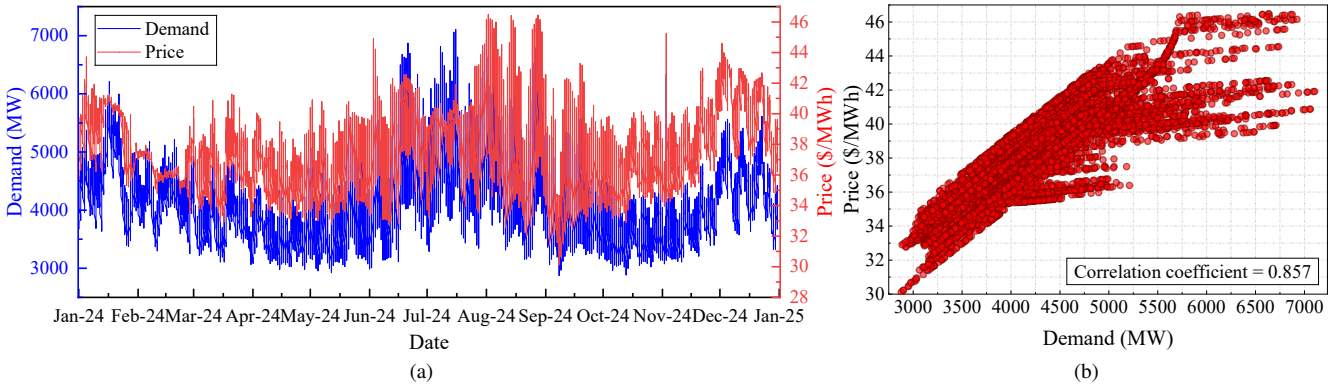

Fig. 3. Training dataset characteristics: (a) hourly demand and electricity price in 2024; (b) demand–price correlation in 2024.

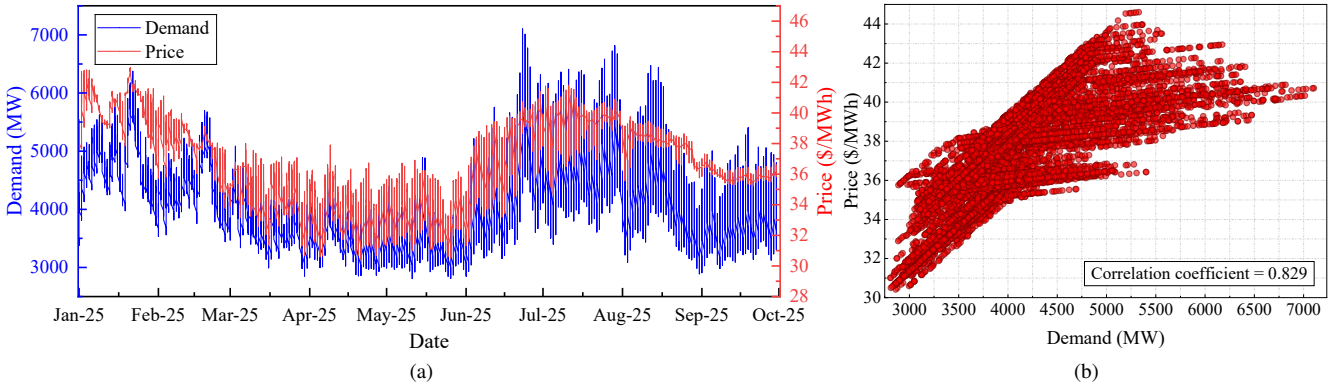

Fig. 4. Testing dataset characteristics: (a) hourly demand and electricity price in 2025; (b) demand–price correlation in 2025.

TABLE III
TESTING PERFORMANCE OF PRICE PREDICTION MODELS USING DEMAND AND ELECTRICITY-PRICE HISTORICAL FEATURES.

Model	RMSE	MAE	R^2
LightGBM	0.3293	0.2504	**0.9851**
RandomForest	0.3476	0.2654	0.9834
XGBoost	0.3493	0.2701	0.9833
GradientBoosting	0.3631	0.2889	0.9819
SVR	0.3741	0.2708	0.9808

LightGBM-estimated price to compute local KKT updates. Across all cases, the recovered operating points remain close to the secure baseline, substantially reducing efficiency loss.

Fig. 5 shows total-cost trajectories versus iterations. In insecure DED, attacks introduce oscillations driven by persistent imbalance and converge to higher costs. With the proposed scheme, convergence remains smooth and terminates near the secure optimum.

C. Scalability Evaluation

The dominant cryptographic operations in the proposed scheme are ECSM and ECDSA signing and verification. On an AMD Ryzen 7 4700U @ 2.0 GHz CPU with 16 GB RAM, the measured latencies for ECSM, ECDSA signing, and ECDSA verification are 0.49 ms, 0.50 ms, and 0.45 ms, respectively. Since the costs are comparable and the Schnorr protocol involves only ECSM, we express computation in units of the ECSM time t_{sm}.

Each DED iteration consists of $4n$ authenticated messages, including supplier uplinks, price broadcasts, partial Schnorr responses, and a final threshold-signed update. The total computational latency per iteration is upper bounded by $(4n+5)\,t_{\mathrm{sm}}$. Based on the protocol payloads (compressed EC points and fixed-point scalars), the average message size is approximately 120 bytes. Given a communication bandwidth $\mathcal{B}$, the overall per-iteration latency is

$$(4n+5)\,t_{\mathrm{sm}} + \frac{2(n+1)\times 120}{\mathcal{B}},$$

which scales linearly with the number of suppliers and is suitable for real-time DED.

Fig. 6 compares the proposed secure DED scheme with conventional distributed (with coordinator) and decentralized (without coordinator) ED frameworks in terms of communication overhead, measured by the number of messages exchanged per iteration. Conventional DED schemes typically do not incorporate cryptographic mechanisms such as digital signatures and therefore lack intrinsic data integrity guarantees. The proposed scheme enforces this crucial security property through authenticated messaging. Standard

TABLE IV
IEEE 118-BUS DED OUTCOMES UNDER NORMAL OPERATION AND DECEPTION ATTACKS WITH AND WITHOUT PROPOSED SECURITY.

Condition	Attack target	Without security			With proposed security		
		Iterations	Price ($/MWh)	Cost ($/h)	Iterations	Price ($/MWh)	Cost ($/h)
Normal	–	58	39.38	125948	58	39.38	125948
FDI	Top 2	157	40.52	137395	78	39.83	126010
Replay	Top 4	99	40.24	127584	92	40.03	126093
Byzantine	Top 10, Bottom 10	170	42.00	165904	24	40.18	126273
FDI + Replay	Top 5	87	40.92	143362	83	40.05	126130
Privacy + Byzantine	All; Top 10, Bottom 10	47	41.98	185134	44	40.16	126310

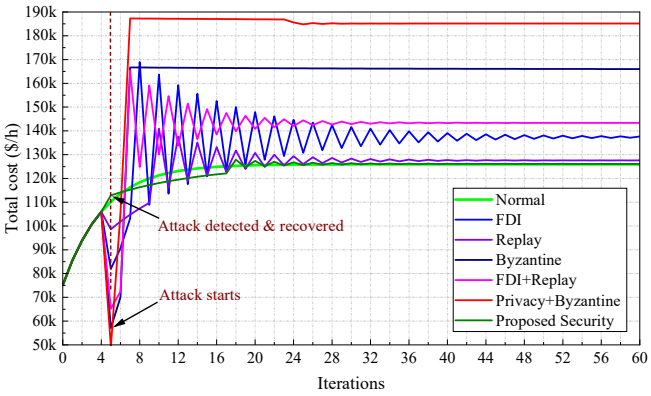

Fig. 5. Cost convergence of DED under normal operation, deception attacks, and secured recovery.

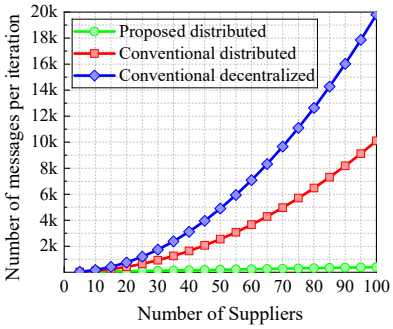

Fig. 6. Scalability comparison of the proposed and conventional DED frameworks in terms of communication overhead.

coordinator-based DED requires $2n$ messages per iteration (uplink n and downlink n). The proposed scheme adds an additional $2n$ messages to enforce global consistency via threshold Schnorr signatures, resulting in at most $4n$ messages per iteration and linear complexity $\mathcal{O}(n)$.

In contrast, coordinator-based DED with P2P cross-verification requires $2n + n(n-1)$ messages per iteration, and fully decentralized ED requires $2n(n-1)$ messages per iteration, both incurring $\mathcal{O}(n^2)$ communication complexity. This quadratic growth increases network congestion and latency, which degrades efficiency and raises feasibility concerns in large-scale, real-time power system optimization.

Since DED algorithms are iterative, convergence typically requires multiple iterations. For a fixed step size, the number of iterations is largely independent of the number of generators for a given optimization algorithm and is therefore excluded from the complexity analysis. Nevertheless, decentralized ED without a coordinator often requires more iterations to converge than coordinator-based DED due to consensus overhead. While the proposed scheme already shows superior scalability in terms of per-iteration overhead, this advantage becomes even more pronounced when considering cumulative overhead and longer propagation delays.

VII. LIMITATION

A key limitation of the coordinator-based DED model is the presence of a single point of failure. This issue is common among many distributed optimization algorithms, such as ADMM, Benders decomposition, and others relying on centralized coordination. While fully decentralized ED models eliminate single points of failure, they are not immune to disruptions. In consensus-based algorithms, the tight coupling of suppliers' computations allows the failure or malicious behavior of a critical supplier to propagate through iterative updates, potentially resulting in suboptimal solutions or convergence failure. It is noteworthy that this paper focuses specifically on deception attacks, where the attacker's goal is not to disrupt or terminate service, but to subtly skew optimization outcomes. These attacks preserve the appearance of convergence while manipulating results to achieve unfair economic gains and suboptimal dispatch.

To address availability concerns, a standby coordinator can be deployed to mitigate denial-of-service attacks and ensure continuity of service.

VIII. CONCLUSION

This paper introduced a scalable, cryptographically verifiable, and robust security scheme to protect distributed economic dispatch from deception attacks, including FDI, replay, and Byzantine manipulation, that degrade the fairness and welfare of electricity markets. Unlike state-of-the-art solutions that focus on isolated threats and rely on inter-supplier communication with quadratic complexity, the proposed approach ensures data integrity, freshness, and consistency with linear complexity, eliminating the need for P2P communication between suppliers. Fault tolerance is achieved through threshold Schnorr signatures, allowing the

system to maintain consistent progress even in the presence of up to $\lfloor (n-1)/3 \rfloor$ malicious suppliers among n participants, while privacy is preserved via lightweight pairwise masking that does not sacrifice optimality. Although centralized coordination inherently supports linear communication complexity, our key contribution lies in retaining this scalability without compromising security guarantees. Beyond deterministic attack detection, we incorporate an ML-based post-attack recovery mechanism so that suppliers can continue operating safely when a manipulation is flagged. Using historical demand and price-derived features, the trained predictor provides an accurate estimate of the clearing price, enabling affected suppliers to compute near-optimal generator output until secure coordination is restored. Case studies on the IEEE 118-bus system demonstrate that the combined cryptographic and ML-guided defense promptly identifies manipulations and substantially limits the resulting welfare loss. Future work will extend these guarantees to more complex market operations, such as security-constrained dispatch and unit commitment.

REFERENCES

[1] N. Tatipatri and S. L. Arun, "A comprehensive review on cyber-attacks in power systems: Impact analysis, detection, and cyber security," *IEEE Access*, vol. 12, pp. 18 147–18 167, 2024.

[2] T. Krause, R. Ernst, B. Klaer, I. Hacker, and M. Henze, "Cybersecurity in power grids: Challenges and opportunities," *Sensors*, vol. 21, no. 18, p. 6225, Sep. 2021.

[3] Sustainalytics, "The downside of digital transformation for utilities–data privacy and cybersecurity risks," https://www.sustainalytics.com/esg-research/resource/investors-esg-blog/the-downside-of-digital-transformation-for-utilities--data-privacy-and-cybersecurity-risks, 2024, accessed: Mar. 17, 2025.

[4] International Energy Agency (IEA), "Enhancing cyber resilience in electricity systems," https://www.iea.org/reports/enhancing-cyber-resilience-in-electricity-systems, 2021, accessed: Mar. 17, 2025.

[5] S. Dareen and V. Srivastava, "Cyberattacks on U.S. utilities surged 70% this year, says Check Point," https://www.reuters.com/technology/cybersecurity/cyberattacks-us-utilities-surged-70-this-year-says-check-point-2024-09-11/, Sep. 2024, accessed: Mar. 17, 2025.

[6] S. Chen, L. Zhang, Z. Yan, and Z. Shen, "A distributed and robust security-constrained economic dispatch algorithm based on blockchain," *IEEE Trans. Power Syst.*, vol. 37, no. 1, pp. 691–700, Jan. 2022.

[7] Z. Wang, G. Chen, and Z. Y. Dong, "Resilient distributed economic dispatch of smart grids under deception attacks," *Nonlinear Dynamics*, vol. 112, pp. 5421–5438, Feb. 2024.

[8] P. Li, Y. Liu, H. Xin, and X. Jiang, "A robust distributed economic dispatch strategy of virtual power plant under cyber-attacks," *IEEE Trans. Ind. Inform.*, vol. 14, no. 10, pp. 4343–4352, Oct. 2018.

[9] B. Huang, Y. Li, F. Zhan, Q. Sun, and H. Zhang, "A distributed robust economic dispatch strategy for integrated energy system considering cyber-attacks," *IEEE Trans. Ind. Inform.*, vol. 18, no. 2, pp. 880–890, Feb. 2022.

[10] Y. Yan, Z. Chen, V. Varadharajan, M. J. Hossain, and G. E. Town, "Distributed consensus-based economic dispatch in power grids using the Paillier cryptosystem," *IEEE Trans. Smart Grid*, vol. 12, no. 4, pp. 3493–3502, Jul. 2021.

[11] W. Chen and G.-P. Liu, "Privacy-preserving consensus-based distributed economic dispatch of smart grids via state decomposition," *IEEE/CAA J. Autom. Sinica*, vol. 11, no. 5, pp. 1250–1261, May 2024.

[12] J. Y. Siu, N. Kumar, and S. K. Panda, "Command authentication using multiagent system for attacks on the economic dispatch problem," *IEEE Trans. Ind. Appl.*, vol. 58, no. 4, pp. 4381–4393, Aug. 2022.

[13] M. Xing, D. Ma, T. Wang, and X. Wang, "Byzantine-resilient distributed algorithm for economic dispatch: A trust-based weight allocation mechanism," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 71, no. 12, pp. 4914–4918, Dec. 2024.

[14] S. Jena, N. P. Padhy, and A. K. Srivastava, "On securing the global economical dispatch in DC microgrid clusters: An event-driven approach," *IEEE Trans. Autom. Sci. Eng.*, vol. 21, no. 4, pp. 6758–6773, Oct. 2024.

[15] Y. Zhang, M. Ni, and Y. Sun, "Fully distributed economic dispatch for cyber-physical power system with time delays and channel noises," *J. Mod. Power Syst. Clean Energy*, vol. 10, no. 6, pp. 1472–1481, Nov. 2022.

[16] A. Kargarian *et al.*, "Toward distributed/decentralized DC optimal power flow implementation in future electric power systems," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 2574–2594, Jul. 2018.

[17] F. Marzbani and A. Abdelfatah, "Economic dispatch optimization strategies and problem formulation: A comprehensive review," *Energies*, vol. 17, no. 3, p. 550, Jan. 2024.

[18] P. K. Jena, S. Ghosh, and E. Koley, "A binary-optimization-based coordinated cyber-physical attack for disrupting electricity market operation," *IEEE Syst. J.*, vol. 15, no. 2, pp. 2619–2629, Jun. 2021.

[19] J. Giraldo, A. Cárdenas, and N. Quijano, "Integrity attacks on real-time pricing in smart grids: Impact and countermeasures," *IEEE Trans. Smart Grid*, vol. 8, no. 5, pp. 2249–2257, Sep. 2017.

[20] S. Tan, W.-Z. Song, M. Stewart, J. Yang, and L. Tong, "Online data integrity attacks against real-time electrical market in smart grid," *IEEE Trans. Smart Grid*, vol. 9, no. 1, pp. 313–322, Jan. 2018.

[21] H. Xu, Y. Lin, X. Zhang, and F. Wang, "Power system parameter attack for financial profits in electricity markets," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3438–3446, Jul. 2020.

[22] M. Esmalifalak, H. Nguyen, R. Zheng, L. Xie, L. Song, and Z. Han, "A stealthy attack against electricity market using independent component analysis," *IEEE Syst. J.*, vol. 12, no. 1, pp. 297–307, Mar. 2018.

[23] G. Liang, S. R. Weller, F. Luo, J. Zhao, and Z. Y. Dong, "Generalized FDIA-based cyber topology attack with application to the australian electricity market trading mechanism," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 3820–3829, Jul. 2018.

[24] D. Hankerson, A. Menezes, and S. Vanstone, *Guide to Elliptic Curve Cryptography*. New York, NY, USA: Springer-Verlag, 2004.

[25] C. P. Schnorr, "Efficient signature generation by smart cards," *Journal of Cryptology*, vol. 4, pp. 161–174, 1991.

[26] Y. Zhang *et al.*, "An optimal combining attack strategy against economic dispatch of integrated energy system," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 70, no. 1, pp. 246–250, Jan. 2023.

[27] X. Huo, H. Huang, K. R. Davis, H. V. Poor, and M. Liu, "A review of scalable and privacy-preserving multi-agent frameworks for distributed energy resources," *Adv. Appl. Energy*, vol. 17, p. 100205, 2025.

[28] M. Qu, "SEC 2: Recommended elliptic curve domain parameters," Certicom Research, Mississauga, ON, Canada, Tech. Rep. SEC2-Ver-1.0, 1999.

[29] T. P. Pedersen, "A threshold cryptosystem without a trusted party," in *Advances in Cryptology—EUROCRYPT '91*, ser. LNCS, vol. 547. Springer, 1991, pp. 522–526.

[30] M. R. Mengis and A. Tajer, "Data injection attacks on electricity markets by limited adversaries: Worst-case robustness," *IEEE Trans. Smart Grid*, vol. 9, no. 6, pp. 5710–5720, Nov. 2018.

[31] C. Liu, M. Zhou, J. Wu, C. Long, and D. Kundur, "Financially motivated FDI on SCED in real-time electricity markets: Attacks and mitigation," *IEEE Trans. Smart Grid*, vol. 10, no. 2, pp. 1949–1959, Mar. 2019.

[32] B. Seshasai, E. Koley, P. K. Jena, and S. Ghosh, "Design of real-time false data injection attack on electricity market with limited sensor accessibility," *IEEE Syst. J.*, vol. 18, no. 4, pp. 1999–2009, Dec. 2024.

[33] Q. Zhang, F. Li, Q. Shi, K. Tomsovic, J. Sun, and L. Ren, "Profit-oriented false data injection on electricity market: Reviews, analyses, and insights," *IEEE Trans. Ind. Inform.*, vol. 17, no. 9, pp. 5876–5886, Sep. 2021.

[34] D. Mishchenko, I. Oleinikova, L. Erdődi, and B. R. Pokhrel, "Multidomain cyber-physical testbed for power system vulnerability assessment," *IEEE Access*, vol. 12, pp. 38 135–38 149, 2024.

[35] P. Wlazlo *et al.*, "Man-in-the-middle attacks and defence in a power system cyber-physical testbed," *IET Cyber-Phys. Syst. Theory Appl.*, vol. 6, no. 3, pp. 164–177, 2021.