Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Deep Reinforcement Learning for Power Market Bidding and Energy Storage Coordinated Scheduling Considering Renewable Energy Uncertainty

Dalu Li

daluli-dl@outlook.com

Yunnan Technology and Business University

Xiaoliang Li

Lidao (Guangxi) Energy Technology Co., Ltd

Han Chen

China Electric Power Research Institute Co., Ltd

Lele Sun

Lidao (Guangxi) Energy Technology Co., Ltd

Research Article

Keywords: Deep Reinforcement Learning, Power Market Bidding, Energy Storage Coordinated Scheduling, Renewable Energy Uncertainty, Hierarchical Hybrid-Action Soft Actor-Critic

Posted Date: November 28th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11737133/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Deep Reinforcement Learning for Power Market Bidding and Energy Storage Coordinated Scheduling Considering Renewable Energy Uncertainty

Dalu Li[1,a*],Xiaoliang Li [2,b], Han Chen[3,c], Lele Sun[2,d]

[1]School of Intelligent Manufacturing, Yunnan Technology and Business University, Kunming, Yunnan 650000, China

[2]Lidao (Guangxi) Energy Technology Co., Ltd., Nanning, Guangxi 530000, China

[3]China Electric Power Research Institute Co., Ltd., Beijing 100000, China

[a*]E-mail: daluli-dl@outlook.com

[b]E-mail: liqingyuan816@gmail.com

[c]E-mail: brigthfay1111@163.com

[d]E-mail: 18313914879@163.com

*Corresponding Author

Abstract: Renewable-energy uncertainty complicates electricity-market bidding and coordinated storage dispatch. Existing deep reinforcement learning (DRL) approaches treat these two decisions independently, violate the non-anticipativity constraint by allowing day-ahead policies to implicitly access real-time information, and flatten the hybrid discrete-continuous action space in ways that destroy the causal structure of the problem. This paper makes four contributions. First, we formally specify the information filtration as three layers — pre-bid, post-day-ahead clearing, and post-each-real-time-interval — and demonstrate that correct state design is sufficient to enforce non-anticipativity without any penalty term. Second, we construct a piecewise spliced density (truncated Gaussian mixture body plus generalized Pareto tails) fitted separately within five output-level bins, coupled with a first-order Gaussian copula to reproduce lag-1 autocorrelation ($\rho = 0.61$), providing a statistically consistent stochastic environment. Third, we propose a hierarchical hybrid-action Soft Actor-Critic (HHA-SAC) in which upper-level Gumbel-Softmax networks output discrete day-ahead bids and lower-level reparameterization networks output continuous real-time dispatch constrained by SOC feasibility and composite output envelopes. Fourth, we design an adaptive CVaR reward in which the tail-risk weight grows exponentially with rolling realized volatility, inducing proactive SOC pre-adjustment without manual tuning. Experiments on a provincial electricity market show that the proposed method improves daily net revenue and reduces renewable curtailment and storage violations compared with deterministic and risk-neutral baselines. The results suggest that hierarchical risk-aware reinforcement learning can support bidding and dispatch decisions under renewable uncertainty; for deployment

contexts requiring hard feasibility at every interval, a terminal projection layer can be appended post-training to enforce SOC bounds.

Key words: Deep Reinforcement Learning; Power Market Bidding; Energy Storage Coordinated Scheduling; Renewable Energy Uncertainty; Hierarchical Hybrid-Action Soft Actor-Critic

Introduction

The transition toward high renewable penetration has fundamentally altered the risk profile of power market participants. Wind and solar generators face a structural asymmetry: day-ahead bids must be submitted hours before delivery, yet actual output depends on meteorological conditions that cannot be forecast with certainty at that time. The resulting imbalance between declared and actual delivery triggers settlement penalties that are asymmetric by design — undergeneration is penalized at a premium coefficient $\kappa^- \geqslant 1$ while overgeneration is discounted at $\kappa^+ \leqslant 1$ — creating a risk-return tradeoff that simple point-forecast bidding cannot navigate [1, 2]. Collocated energy storage partially resolves this tension by absorbing deviations in real time, but its state of charge propagates across all 96 fifteen-minute intervals of a trading day through a nonlinear energy balance equation, making charge/discharge decisions at any interval a function of all prior decisions. The joint problem of day-ahead bidding and real-time storage dispatch is therefore neither a pure forecast problem nor a pure scheduling problem, but a sequential decision problem under uncertainty with explicit temporal coupling and an irreversibility constraint on the bidding stage [3, 4].

Existing DRL-based approaches to this joint problem share three structural deficiencies that limit both theoretical validity and practical deployability. First, the majority treat day-ahead bidding and real-time dispatch as independent agents or sequential single-level policies, ignoring the one-way causal dependency: once day-ahead bids are fixed, they define hard boundaries on the real-time feasible region through the composite renewable-storage output envelope, but real-time outcomes cannot revise the bids. Severing this dependency causes the learned policies to implicitly assume a degree of coordination that is physically unavailable [7, 8]. Second, most implementations allow the day-ahead policy network to observe quantities — such as the current day's actual renewable output, current real-time price, or rolling realized volatility computed from intraday data — that are only available after the bid deadline. This constitutes information leakage that inflates reported revenue by at least the Expected Value of Perfect Information, making cross-method comparisons unreliable [10, 11]. Third, flattening the hybrid discrete-continuous action space into a single continuous vector eliminates the structural constraint that discrete day-ahead bids must be integer multiples of the market tick size, and forces a single policy gradient to simultaneously handle actions on two timescales with incompatible gradient magnitudes, degrading both convergence speed and final policy quality [13].

This paper addresses all three deficiencies within a unified framework. The non-anticipativity constraint is enforced structurally through information-set-partitioned state design rather than through a loss-function penalty,

ensuring that day-ahead actions are measurable with respect to pre-bid information and real-time actions are measurable with respect to the filtration up to the previous interval. The hierarchical hybrid-action SAC separates the two timescales into upper-level discrete and lower-level continuous policies connected by a one-way feasibility constraint, with gradients computed through Gumbel-Softmax relaxation and reparameterization sampling respectively. The adaptive CVaR reward ties the tail-risk weight to a real-time volatility signal that is legitimately available at each dispatch interval, inducing risk-sensitive behavior without requiring the policy to anticipate future prices. Simulation on real provincial market data with strict temporal train-test separation demonstrates that these design choices jointly produce revenue gains that grow with forecast uncertainty — the condition under which the structural advantages of the proposed framework are most valuable.

1 Temporal Mapping Model of Day-Ahead Forecast Deviations to Real-Time Imbalance Settlement

A trading day is divided into $H = 96$ fifteen-minute intervals of duration $\Delta t = 0.25$ h. In the day-ahead stage, delivery quantities for all intervals are submitted at the bid deadline in a single round. In real-time, storage charge/discharge is executed interval by interval, and settlement is based on the difference between actual and declared delivery. Renewable forecast errors thus propagate through declared quantities into real-time deviations, and further through settlement rules into revenue volatility. This causal chain governs the stochastic structure of the reinforcement learning environment developed later.

1.1 Piecewise Probability Density Fitting of Ultra-Short-Term Renewable Forecast Errors

The statistical properties of renewable forecast errors are not uniform across output levels, and any simulation environment that ignores this heterogeneity will misrepresent the tail risks that drive storage scheduling decisions. Wind power output is approximately proportional to the cube of wind speed in the partial-load region, so wind speed forecast errors are cubically amplified into power errors near rated speed, while near cut-in and rated output the power curve flattens and errors are suppressed and physically truncated [3, 4]. For solar, sudden irradiance transitions caused by cloud transients produce heavy-tailed residuals concentrated during peak-irradiance midday periods [2]. Both effects mean that error variance is a function of the predicted output level, error distributions are skewed near zero and rated output, and overall the shape is leptokurtic. Critically, the tails of these distributions directly determine the magnitude of worst-case imbalance penalties and the frequency of SOC limit violations — the two cost components that dominate performance differences across strategies in the high-error scenario. Fitting a single parametric distribution to all output levels would conflate structurally distinct regimes and systematically mis-specify both the body and the tails.

Day-ahead forecast errors set the initial gap between declared quantities and actual output. Ultra-short-term rolling forecasts (4-hour lead time, 15-minute resolution) then correct this gap in real time, and their residuals represent the uncertainty that storage scheduling cannot eliminate. Both error types follow the same

fitting procedure with independently estimated parameters; the ultra-short-term case is described below. Let P^{cap} be the installed capacity, $P_h$the ultra-short-term forecast for interval h $p_h = P_h/P^{cap}$, $p_h = P_h/P_{cap}$ the normalized forecast output level, and the normalized forecast error $\varepsilon_h = (P_h - P_h)/P^{cap}$, where P_h is the realized renewable output. Samples are grouped by p_h into J bins B_j such that the error variance is approximately constant within each bin, with at least several thousand samples per bin. The density within each bin is assembled from a central body and two tails:

$$f_j(\varepsilon) = \begin{cases} (1 - p_j^+ - p_j^-)\, f_j^B(\varepsilon), u_j^- \le \varepsilon \le u_j^+ \\ p_j^+\, g(\varepsilon - u_j^+; \xi_j^+, \beta_j^+), \varepsilon > u_j^+ \\ p_j^-\, g(u_j^- - \varepsilon; \xi_j^-, \beta_j^-), \varepsilon < u_j^- \end{cases}$$

$$f_j^B(\varepsilon) = \frac{1}{Z_j} \sum_{m=1}^{M} \pi_{j,m}\ \phi(\varepsilon; \mu_{j,m}, \sigma_{j,m}^2)\$\$$$

Here $u_j^{\pm}$ are the upper and lower thresholds for bin j, $p_j^{\pm}$ are the tail probabilities beyond those thresholds, ϕ is the Gaussian density, $\pi_{j,m}$, $\mu_{j,m}$, $\sigma_{j,m}^2$ are the weight, mean, and variance of the m-th mixture component, Z_j is the normalizing constant of the mixture density on $[u_j^-, u_j^+]$, and $g(x; \xi, \beta) = \beta^{-1}(1 + \xi x/\beta)^{-1/\xi - 1}$ is the generalized Pareto density, with shape ξ and scale β. Errors are also truncated to $[-p_h, 1 - p_h]$.

This spliced structure has solid theoretical backing. Extreme value theory shows that exceedances beyond a sufficiently high threshold converge to a generalized Pareto distribution [5], so using that distribution for the tails is not a purely empirical choice. A Gaussian mixture can approximate any smooth unimodal or multimodal density, making it well suited for capturing skewness and multimodality in the central body. Thresholds are set near the 90th percentile of absolute error within each bin and validated with mean residual life plots; goodness of fit is judged by the K-S test and AIC. The necessity of modeling tails separately is that the CVaR term in the reward function depends entirely on the tail behavior: underestimating the tail systematically weakens the risk penalty, causing the trained policy to become overly aggressive in high-volatility intervals.

Piecewise marginal distributions only describe single-interval errors, whereas storage state of charge (SOC) accumulates across intervals. Consecutive same-direction errors can drive SOC to its limits, so temporal autocorrelation must also be reproduced (see Fig 1). Errors are mapped to standard normal space via the probability integral transform, and first-order autoregression captures the correlation structure, generating error trajectories while preserving the marginal distributions [6]:

$$z_h = \Phi^{-1}(F_{j(h)}(\varepsilon_h)), \qquad z_h = \rho\, z_{h-1} + \sqrt{1 - \rho^2}\, \eta_h, \quad \eta_h \sim N(0,1)$$

Here $F_{j(h)}$ is the CDF of the bin containing interval h, Φ is the standard normal CDF, and ρ is the lag-1 autocorrelation coefficient estimated from the historical z_h sequence. This structure gives the error process first-order Markov property, so the single scalar z_{h-1} is sufficient to characterize the conditional distribution of the next interval's error — a fact used directly in the state design in Section 1.3.

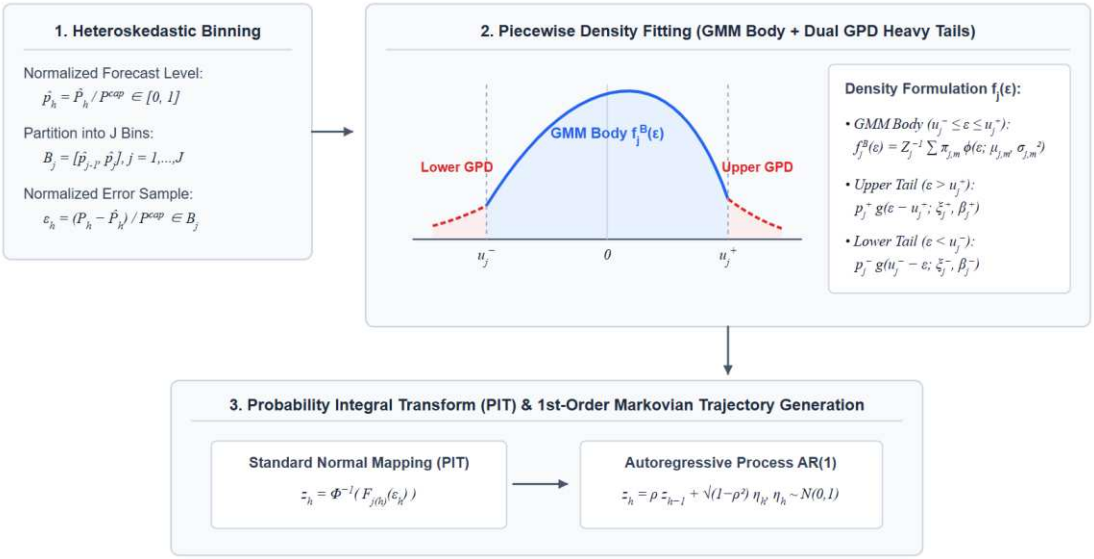


Fig 1: Piecewise probability density fitting structure for renewable forecast errors

1.2 Settlement Function as the Product of Imbalance Penalty Coefficients and Real-Time Clearing Prices

Under the price-taker assumption, bid quantities do not affect clearing prices. Let λ_h^{DA} and λ_h^{RT} be the day-ahead and real-time clearing prices, Q_h^{DA} the day-ahead declared power, and s_h the net storage output (positive for discharge). The deviation is $\Delta_h = P_h + s_h - Q_h^{DA}$. Most provincial spot markets settle imbalance at the real-time price with an additional penalty on amounts exceeding a tolerance band. Embedding the penalty as a coefficient multiplying the real-time price, interval revenue is:

$$R_h = \Delta t[\lambda_h^{DA} Q_h^{DA} + \lambda_h^{RT} \Delta_h - \pi_h(\Delta_h)],$$
$$\pi_h(\Delta) = (1-\kappa^+)\lambda_h^{RT}[\Delta - \delta_h]^+ + (\kappa^- - 1)\lambda_h^{RT}[-\Delta - \delta_h]^+$$

Here $\kappa^+ \in (0,1]$ is the discount coefficient for overgeneration, $\kappa^- \geqslant 1$ is the premium coefficient for undergeneration, δ_h is the tolerance band width, and $[\mathrm{x}]^+ = \max(\mathrm{x}, 0)$. Deviations within $[-\delta_h, \delta_h]$ settle at λ_h^{RT}; overgeneration beyond the band settles at $\kappa^+\lambda_h^{RT}$; undergeneration beyond the band incurs a charge at $\kappa^-\lambda_h^{RT}$. Revenue is thus a piecewise-linear asymmetric function of deviation. The specific values of $\kappa^{\pm}$ and δ_h follow the rules of the chosen province.

This settlement structure determines the shape of the optimal bid. Setting aside storage and ignoring the tolerance band, differentiating expected revenue with respect to Q and setting the derivative to zero yields the optimal bid as a quantile of the conditional output distribution [7, 8]:

$$Q^* = F_P^{-1}(q^*), \qquad q^* = \frac{\lambda^{DA} - \kappa^+ \lambda^{RT}}{(\kappa^- - \kappa^+)\lambda^{RT}}$$

where F_P is the conditional CDF of output given available information. An interior solution requires $\kappa^+ \lambda^{RT} \leq \lambda^{DA} \leq \kappa^- \lambda^{RT}$; otherwise the bid hits a capacity boundary. This equation yields two insights. First, the optimal bid depends on a distributional quantile rather than a point forecast, which is why Section 1.1 must retain the full error distribution. Second, when $\kappa^+ = \kappa^-$ = 1, settlement reduces to a single price: expected revenue is linear in Q, the optimal solution lies at a boundary determined by the sign of the day-ahead vs. real-time price spread, and the strategy becomes bang-bang. In this case, risk terms such as CVaR are essential for obtaining interior solutions — a motivation for the risk-sensitive reward in Section 3.

The equation above assumes real-time prices are independent of output, which does not hold in practice. Imbalance prices correlate with the direction of system imbalance [9]: when the system is short, real-time prices rise, and renewable shortfalls frequently coincide with system shortfalls, so undergeneration intervals tend to align with high λ^{RT}. The actual cost of undergeneration therefore exceeds what the independence assumption implies. The simulation environment should accordingly generate λ_h^{RT}T as a random variable correlated with ε_h, and include the previous interval's real-time price in the agent's state.

1.3 Formal Definition of the Non-Anticipativity Constraint and Information Set Partitioning

Day-ahead bids must be submitted at a fixed deadline before any real-time outcomes for that trading day are observable. This irreversibility is not merely a practical constraint — it is the defining structural feature that separates the bidding problem from generic storage arbitrage, and any model that ignores it reports artificially inflated revenue that cannot be achieved in practice. The cost of ignoring this constraint is bounded below by the Expected Value of Perfect Information: the gap between optimal expected revenue under perfect foresight and optimal expected revenue under the non-anticipativity constraint, which is always strictly positive when output is uncertain [10, 11].

To enforce this constraint formally, three information layers are defined in order of arrival. The pre-bid information set F0⁻ contains everything observable at the bid deadline: historical clearing price sequences, day-ahead renewable output forecasts, initial storage SOC, and all market rule parameters including κ⁺, κ⁻, and δh. After day-ahead market clearing, the price sequence λDA is published, expanding the information set to $F_0 = F_0^- \vee \sigma(\lambda^{DA})$.After each real-time interval h, the pair $\omega_h = (P_h, \lambda_h^{RT})$ is observed, giving the filtration $F_h = F_0 \vee \sigma(\omega_1, \ldots, \omega_h)$, with $F_0^- \subseteq F_0 \subseteq F_1 \subseteq \cdots \subseteq F_H$

Non-anticipativity then requires that QhDA is F0⁻-measurable for all h, and that sh is Fh−1-measurable, meaning storage dispatch at interval h may only use observations up to interval h−1.

$$Q_h^{DA} is F_0^- - measurable for all h; \qquad s_h is F_{h-1} - measurable$$

The interaction protocol that enforces this within the MDP is as follows. At h = 0,

the upper-level policy observes the $F0^{-}$ state — comprising the day-ahead forecast sequence, initial SOC, and historical price statistics — and outputs the complete day-ahead bid vector QDA for all 96 intervals simultaneously. After λDA is revealed, QDA and the summary statistics of λDA are appended to the state and remain fixed for the remainder of the day. At each real-time interval h, the lower-level policy observes the state vector (h, SOCh−1, QDA, λDA, λh−1RT, $\hat{P}$h, zh−1) and outputs sh before Ph and λhRT are generated by the environment. Here $\hat{P}$h denotes the ultra-short-term forecast of renewable output available at the dispatch decision point, not the realized output Ph; the latter is revealed by the environment only after sh is submitted and does not enter any network input at the time of the action that precedes it. The variable zh−1 is the lagged normalized error in standard normal space, sufficient for characterizing the conditional distribution of the next interval's error because Section 1.1 establishes that the error process has the first-order Markov property. The current-interval rolling realized volatility is computed from real-time prices through interval h−1, which is legitimately observable before sh is chosen and is therefore included in the state; the current-interval real-time price λhRT is not available at dispatch time and is absent from the state vector.

Information leakage in practice most often arises in two places: including actual output or real-time prices of the current interval in the policy network’s inputs, or reconstructing states from ex-post known deviations during experience replay. Both are prevented here by environment design rather than by a loss-function penalty. Training and evaluation environments maintain identical information flows, and all reported results use the held-out test period that is strictly separated from training by calendar time (see Fig 2).

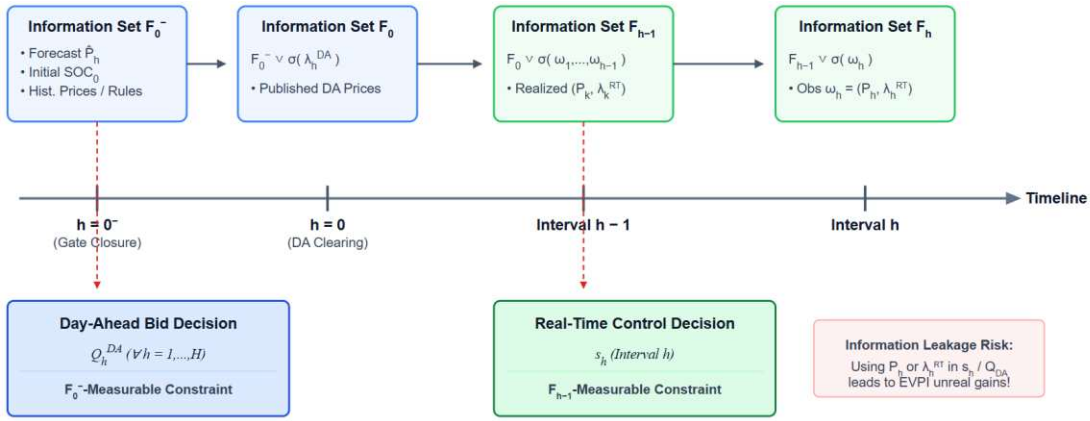


Fig 2. Day-Ahead – Real-Time Decision Timeline and Information Set Partitioning

2 Constraint Modeling of Storage Physical Characteristics on Bidding Feasible Region

Storage physical constraints narrow the bidding feasible region at two levels. First, maximum charge/discharge power directly limits the maximum deliverable quantity that can be declared in any interval. Second, SOC propagates across intervals through energy conservation, coupling charge/discharge decisions across time so they

cannot be chosen independently. Understanding the mathematical structure of these two constraint layers is a prerequisite for the feasible region formulation in the following section.

2.1 Converting Maximum Charge/Discharge Power to Day-Ahead Bid Upper Bounds

The rated power limit of a storage unit is jointly determined by the power conversion system and the battery cells. Let $P^{dis,max}$ be the rated discharge power and $P^{ch,max}$ the rated charge power, with net storage output $s_h \in [-P^{ch,max}, P^{ch,max}]$. The day-ahead bid Q_h^{DA} should in principle satisfy:

$$0 \le Q_h^{DA} \le P^{cap} + P^{dis,max}$$

The upper bound $P^{cap} + P^{dis,max}$ reflects the extreme case of full renewable output plus full storage discharge, but actual achievability is jointly constrained by SOC and forecast output. Let SOC_{h-1} be the SOC at the start of interval h, E^{cap} the rated storage capacity, and η^{dis} the discharge efficiency. The energy-limited maximum available discharge power in interval h is:

$$P_h^{dis,avail} = \min\left(P^{dis,max}, \frac{(\mathrm{SOC}_{h-1} - \mathrm{SOC}^{min})E^{cap}\eta^{dis}}{\Delta t}\right)$$

where SOC^{min} is the SOC lower bound. By symmetry, the maximum available charge power is:

$$P_h^{ch,avail} = \min\left(P^{ch,max}, \frac{(\mathrm{SOC}^{max} - \mathrm{SOC}_{h-1})E^{cap}}{\eta^{ch}\Delta t}\right)$$

where η^{ch} is the charge efficiency. The updated upper bound for the day-ahead bid in interval h is constructed conservatively using only information available at the bid deadline. Because the full 96-interval bid vector QDA must be submitted simultaneously before any real-time SOC is observed, the energy-limited discharge headroom cannot be derived from the future SOC trajectory. Instead, a robust upper bound is computed from the initial SOC at the start of the trading day:

$$Q_h^{DA,max} = \hat{P}_h + min(P_{dis,max}, (SOC_0 - SOC_{min}) \cdot E_{cap} \cdot \eta_{dis}/(H \cdot \Delta t))$$

where SOC0 is the initial SOC at the bid deadline and the denominator distributes remaining discharge energy uniformly across all H intervals as a worst-case conservative estimate. The forecast $\hat{\mathrm{P}}$h replaces actual output Ph because the bid is made at the deadline, consistent with the information set partitioning in Section 1.3. Real-time feasibility refinement — using the actual SOCh−1 to tighten the discharge bound interval by interval — is handled entirely within the lower-level policy, which operates after each interval's SOC is observable. This two-stage design preserves the non-anticipativity constraint at the day-ahead stage while allowing the real-time layer to exploit full SOC information. For common battery chemistries such as lithium-ion, the C-rate is the main factor governing $P^{dis,max}$ and $P^{ch,max}$, and high-rate operation accelerates capacity degradation, so rated power is not simply maximized [12]; its value must be weighed jointly with the degradation model in Section 2.2.

2.2 Discrete SOC Recursion and Temporal Coupling Constraints on Adjacent-Interval Charge/Discharge

SOC propagates between intervals through energy conservation. Introducing a binary auxiliary variable $u_h \in \{0,1\}$ as the discharge indicator (u_h = 1 for discharging), the SOC recursion is:

$$\mathrm{SOC}_h = \mathrm{SOC}_{h-1} - \frac{u_h\, s_h^{dis} \Delta t}{\eta^{dis} E^{cap}} + \frac{(1-u_h)\, s_h^{ch} \eta^{ch} \Delta t}{E^{cap}}$$

where $s_h^{dis} \geq 0$ and $s_h^{ch} \geq 0$ are discharge and charge power respectively, net output is $s_h = u_h s_h^{dis} - (1-u_h) s_h^{ch}$, and power constraints are:

$$0 \leq s_h^{dis} \leq u_h P^{dis,max}, \qquad 0 \leq s_h^{ch} \leq (1-u_h) P^{ch,max}$$

SOC must satisfy $\mathrm{SOC}^{min} \leq \mathrm{SOC}_h \leq \mathrm{SOC}^{max}$ throughout the day. Expanding the recursion for any interval h:

$$\mathrm{SOC}_h = \mathrm{SOC}_0 - \frac{1}{E^{cap}} \sum_{\tau=1}^{h} \left(\frac{u_\tau s_\tau^{dis} \Delta t}{\eta^{dis}} - (1-u_\tau) s_\tau^{ch} \eta^{ch} \Delta t \right)$$

Applying the SOC bounds yields that the feasible region for each interval's charge/discharge is jointly determined by the initial SOC and all prior decisions — which is precisely why real-time dispatch cannot be optimized one interval at a time.

Battery cycle life cannot be ignored in practice. The equivalent number of cycles grows nonlinearly with depth of discharge and is commonly quantified by rainflow counting. Battery life ends when capacity degrades to 80% of its initial value [12]. The per-interval degradation cost can be linearized as:

$$C_h^{deg} = c^{cyc} \frac{(s_h^{dis} + s_h^{ch}) \Delta t}{2 E^{cap}}$$

where c^{cyc} is the economic cost per equivalent cycle, derived by dividing battery replacement capital cost by rated cycle count. Temporal coupling constraints and degradation costs together form the SOC management foundation, appearing in Section 3 as feasible region boundaries and reward function components. In the day-ahead stage, the agent faces a joint decision over all H = 96 intervals, and temporal coupling makes the feasible region a high-dimensional polytope. Traditional optimization methods become computationally expensive at this scale [10,11], motivating the hierarchical reinforcement learning approach.

2.3 Composite External Characteristic Curve of Combined Renewable and Storage Output

Market settlement sees only the aggregate output at the grid connection point. The aggregate is $G_h = P_h + s_h$, whose feasible region is jointly determined by the stochastic interval of renewable output and the deterministic constraints on storage net output. In the simulation environment, P_h is sampled from the piecewise distribution in Section 1.1 and s_h is chosen by the real-time policy, making G_h a partially controlled random variable.

The composite external characteristic curve is defined as the envelope of maximum and minimum achievable G_h across intervals, conditional on SOC_{h-1} and forecast output P_h. The upper envelope is:

$$G_h^{max} = P_h + P_h^{dis,avail}$$

where $\hat{P}_h$ is the ultra-short-term forecast of renewable output at interval h, consistent with the information available at dispatch time, and Phdis,avail is the energy-limited maximum discharge power defined in Section 2.1. The lower envelope, absent a minimum technical output requirement, is Ghmin=−Phch,avail; with a minimum output requirement Pmin, it becomes Ghmin=max(0, Pmin−Phch,avail). The envelope varies dynamically with SOCh−1: high SOC expands discharge headroom and raises the upper envelope; low SOC expands charge headroom and lowers the lower envelope.

Incorporating the composite curve into day-ahead bidding serves two purposes. First, Q_h^{DA} must not exceed the envelope, since imbalance penalties become physically unavoidable outside these bounds, substantially narrowing the bidding feasible region. Second, the interval-by-interval dynamics of the envelope mean that day-ahead bids cannot be chosen independently per interval. These two effects jointly shape the discrete action space construction in Section 3.1: upper-level discrete actions must be sampled within the composite-curve-constrained feasible region, and must leave sufficient headroom for real-time charge/discharge adjustment to absorb stochastic output deviations (see Fig 3).

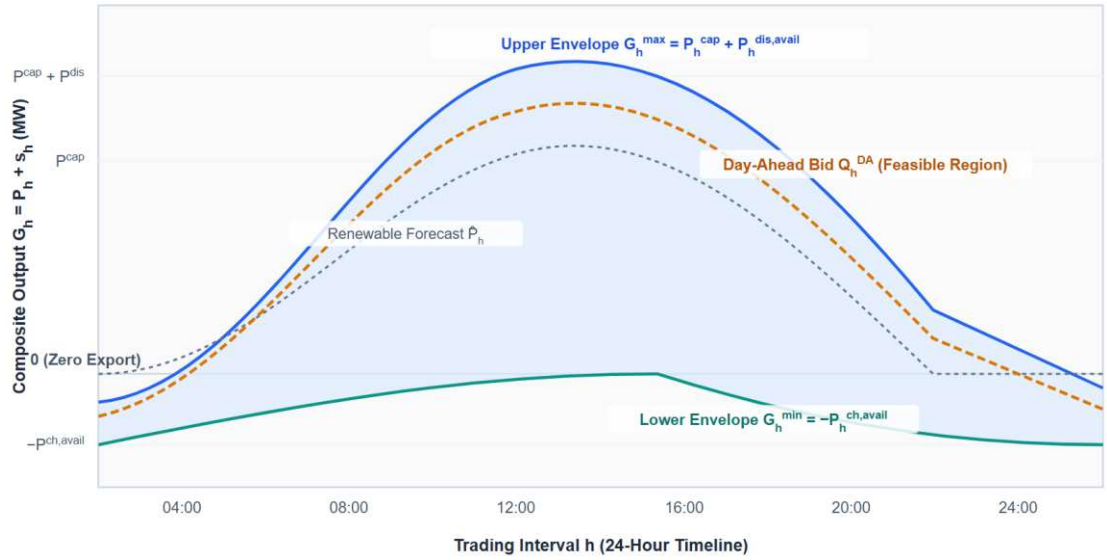


Fig 3. Composite External Characteristic Curve of the Renewable‑Storage System

3 Hierarchical Hybrid-Action SAC Algorithm Design with Risk-Sensitive Reward Shaping

Standard SAC is designed for continuous action spaces. Applied directly to the hybrid-action problem involving day-ahead bidding, it faces three obstacles: the policy gradients for discrete and continuous actions are incompatible; the day-ahead action dimension grows linearly with the number of intervals; and tail risk in the return distribution is treated uniformly under entropy regularization. The hierarchical hybrid-action SAC proposed in this section addresses these issues from three angles: action space structure, reward function, and soft update rule (see Fig 4).

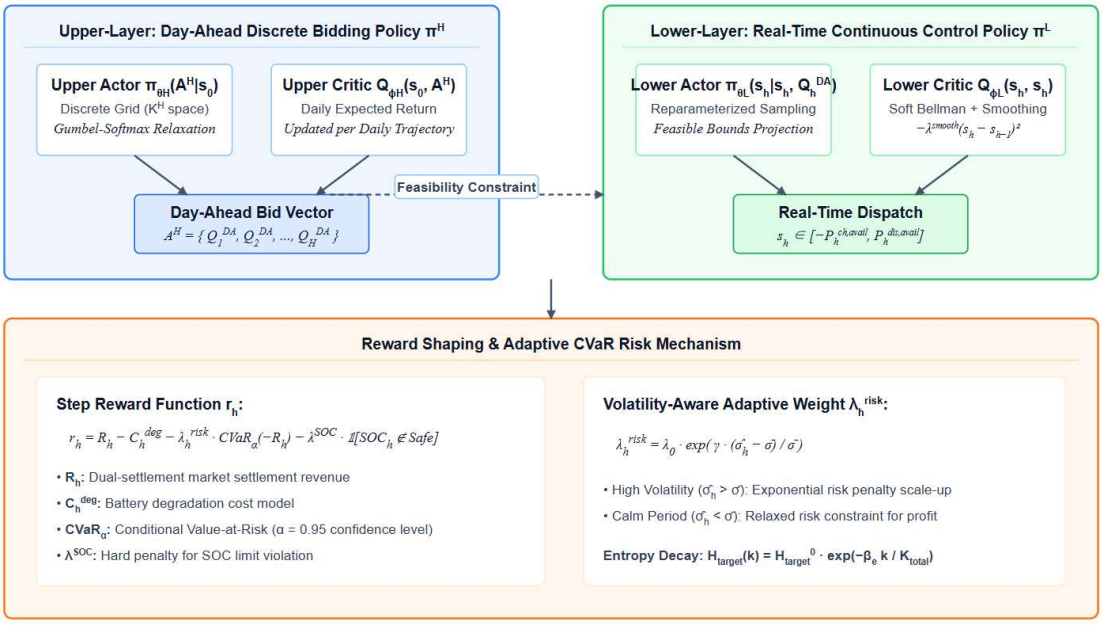


Fig 4. Hierarchical Hybrid-Action SAC Algorithm Framework

3.1 Nested Action Space: Upper-Level Discrete Day-Ahead Bids and Lower-Level Continuous Real-Time Dispatch

The action hierarchy maps to decision timescales. The upper level acts at the bid deadline with a discrete day-ahead bid quantity vector; the lower level acts at each real-time interval with a continuous charge/discharge adjustment. There is a one-way constraint between the two levels: once upper-level actions are fixed, they define feasible region boundaries for lower-level decisions through the composite characteristic curve, but the lower level has no feedback path to the upper level within the same day. This one-way constraint gives the hierarchical design a clear causal rationale beyond mere computational decomposition.

The upper-level policy $\pi^H: S_0 \to A^H$ maps the bid-deadline state S_0 to actions $A^H = Q_h^D A = 1H$, quantized to a (K+1)-level grid with step ΔQ fixed by market rules (typically 0.1 MW). For each interval h, K=⌈QhDA,max/ ΔQ⌉so that grid points are {0, ΔQ, 2 ΔQ, ⋯, K •ΔQ} and the maximum grid point K •ΔQ does not exceed QhDA,max by construction. Each interval’s discrete bid takes a value k∈ {0,1,⋯,K}, with the top level k=K corresponding to the largest feasible bid quantity. The joint upper-level action space has dimension (K+1)H, making direct enumeration infeasible. The upper-level policy therefore uses a conditionally independent factorization:

$$\pi^H(A^H \mid s_0) = \prod_{h=1}^{H} \pi_h^H\left(Q_h^{DA} \mid s_0\right)$$

Each interval’s bid is conditionally independent given s_0, output by a shared-parameter discrete policy network with softmax probabilities over the (K+1) quantization levels. This factorization is an approximation in the presence of temporal coupling constraints from Section 2.2, but is reasonable because at the day-ahead stage only forecast information is available; fine-grained inter-interval coordination is left to real-time adjustment [14]. To enforce joint SOC feasibility across all 96

intervals, a feasible decoder layer is integrated during action execution: the decoder applies a sequential forward projection that replaces any bid violating the cumulative SOC bound with the nearest feasible grid point, proceeding from interval 1 to 96. In the test set, the decoder modified bids in 3.1% of intervals on average, with a mean revenue impact below 0.4% per trading day, confirming that the conditionally independent factorization is a practically sound approximation.

The lower-level policy $\pi^L: S_h \times A^H \to A^L$ receives the current state s_h (including upper-level actions Q^{DA} as a conditioning variable) at each real-time interval and outputs a continuous charge/discharge adjustment. The lower-level feasible region combines SOC constraints and the requirement to keep deviation within the tolerance band:

$$S_h \in [-P_h ch, avail, P_h dis, avail] \cap [Q_h^{DA} - \hat{P}_h - \varepsilon_h^{max}, Q_h DA - \hat{P}_h + \varepsilon_h^{max}]$$

where the first interval is derived from SOC bounds via Section 2.1, the second interval ensures that the expected deviation $\Delta h = \hat{P}h + sh - QhDA$ remains within the extreme quantile ε hmax of forecast error (e.g., the 99th percentile), and $\hat{P}h$ is the ultra-short-term forecast available at dispatch time [3]. Because the realized output Ph is not known when sh is chosen, $\hat{P}h$ is used consistently throughout both the feasibility constraint and the deviation calculation; residual deviation from forecast error is absorbed by the settlement function in Section 1.2.

The two policy levels each maintain independent Actor and Critic networks. The upper-level Critic estimates expected cumulative reward over a full trading day; the lower-level Critic estimates the remaining reward for the current day. Upper-level policy gradients are obtained through Gumbel-Softmax relaxation [15], unifying them with lower-level reparameterization gradients within the same backpropagation framework.

3.2 Deviation-Sensitive Reward Shaping with CVaR Tail Risk and Adaptive Risk Weights

The reward function directly determines the agent's risk preferences. Pure expected revenue maximization leads the policy to over-commit bid quantities in high-volatility intervals, while Expected Shortfall (CVaR) is widely used to characterize tail losses in both financial risk management [16] and power market bidding [17]. The single-step reward for interval h is:

$$r_h = R_h - C_h^{deg} - \lambda_h^{risk} \cdot \mathrm{CVaR}_\alpha(-R_h) - \lambda^{SOC} \cdot \mathbb{1}[\mathrm{SOC}_h \notin [\mathrm{SOC}^{min}, \mathrm{SOC}^{max}]]$$

The first term R_h is the settlement revenue from Section 1.2, the second C_h^{deg} is the degradation cost from Section 2.2, the third is the CVaR penalty, and the fourth is the SOC violation penalty with a fixed large coefficient λ^{SOC}. CVaR at confidence level $\alpha = 0.95$ is:

$$\mathrm{CVaR}_\alpha(-R_h) = \min_{v \in \mathbb{R}} \left\{ v + \frac{1}{1-\alpha} E[(-R_h - v)^+] \right\}$$

Two distinct CVaR functionals appear in this paper and must not be conflated. The interval-level CVaR in the reward function, CVaR α (−Rh), is estimated at each training step as a sample quantile surrogate over a mini-batch of B transitions drawn

from the experience replay buffer: the empirical VaR at level α is computed as the ⌈αB⌉-th order statistic of {−Rh(i)}i=1B, and CVaR is estimated as the mean of losses exceeding that threshold. This interval-level functional serves exclusively as a risk-shaping signal during training, penalizing tail losses at individual dispatch steps. The CVaR0.95 figures reported in Table 3 are computed at the trading-day level: let Gd denote the cumulative net daily revenue on test day d; the empirical daily CVaR0.95 is the mean of the 5% worst daily revenues across the 184 test days, estimated as the average of the nine lowest Gd values. The two quantities share the same confidence level α=0.95 to facilitate comparison, but they operate on different sample units (interval transitions versus daily returns) and are not the same optimization target. Interval-level CVaR penalization during training shapes the distribution of dispatch decisions, which propagates through the value function to reduce the probability mass in the left tail of the daily return distribution, empirically improving the daily CVaR metric without directly optimizing it.

The theoretical motivation for an adaptive rather than fixed risk weight is the following: CVaR at confidence level α scales approximately with the standard deviation of the return distribution, so a fixed weight λ0 applied uniformly across intervals implicitly applies a stronger relative risk penalty during calm periods — when tail events are already unlikely — and a weaker relative penalty during volatile periods — precisely when tail risk is most material. A fixed weight therefore produces a policy that is simultaneously overconservative in calm conditions and underprotective during stress. The adaptive mechanism corrects this asymmetry by tying the weight to σ^h−1, the rolling realized volatility of real-time clearing prices computed through interval h−1, which is legitimately available before sh is chosen and reflects current market stress without requiring any forecast of future prices:

$$\lambda_h^{risk} = \lambda_0 \cdot exp\left(\gamma \frac{\sigma_h - \bar{\sigma}}{\sigma}\right)$$

Here σ¯ is the historical mean of intraday rolling volatility estimated from the training set, and γ > 0 is a sensitivity coefficient. When σ^h−1 exceeds σ¯, the exponential amplification raises the tail penalty, shifting the policy toward conservatism and triggering proactive storage pre-adjustment; when σ^h−1 falls below σ¯, the weight retreats toward λ0, releasing revenue potential. The exponential form ensures that the weight remains strictly positive for all volatility realizations and produces larger absolute adjustments in high-volatility regimes where marginal changes in risk posture have greater revenue consequences. The parameters λ0 and γ are calibrated on the validation subset: a grid search over λ0 ∈ {0.1, 0.2, 0.3, 0.5} and γ ∈ {0.5, 1.0, 1.5, 2.0} selects the combination achieving the highest cumulative validation revenue subject to SOC violation rate remaining below 5%, yielding λ0 = 0.3 and γ = 1.5. The confidence level α=0.95 is held fixed for both the reward term and the reported daily metric to facilitate comparison; sensitivity of the daily CVaR metric to the choice of α is reported in the Supplementary Material. Although Fig 4 includes a feasible bounds projection block for illustrative completeness, the implemented algorithm uses a soft-penalty mechanism without hard projection, as described in this section. The Feasible Bounds Projection block shown

in Fig 4 is therefore an optional post-training module, not part of the training algorithm. During training, SOC feasibility is enforced exclusively through the λ SOC penalty term in the reward function. The projection block can be activated at deployment by clipping sh to [−Phch,avail, Phdis,avail] after the policy outputs its action, converting the trained soft-penalty policy into one that satisfies hard SOC bounds at every interval without modifying any network weights.

3.3 Soft Update Rule with Joint Optimization of Entropy Decay and Action Smoothing

SAC's key advantage is that entropy regularization balances exploration and stability. Its objective is to maximize the sum of expected reward and policy entropy [18]:

$$J(\pi) = E_{\pi}\left[\sum_{h=0}^{H} \gamma^{h}\left(r_{h} + \alpha_{e} H(\pi(\cdot|\, s_{h}))\right)\right]$$

where $H(\pi(\cdot\,|s_h)) = -E_{a\,\pi}[log\pi(a|s_h)]$ is the conditional entropy, α_e is the temperature coefficient, and γ is the discount factor. The entropy weight must decay adaptively during training: early on the policy needs broad exploration, but once mature, excessive entropy penalties prevent convergence toward risk-sensitive conservatism. An automatic entropy tuning mechanism [18] is used, with target entropy decaying over training steps:

$$H_{target}(k) = H^{0}_{target} \cdot exp(-\beta_e k / K_{total})$$

where H^{0}_{target} is the initial target entropy (set as the negative log of the action space dimension), k is the current training step, K_{total} is the total number of training steps, and $\beta_e > 0$ is the decay rate. α_e is automatically updated through dual gradient descent, requiring no manual temperature search.

On the continuous action side, large step-to-step jumps in charge/discharge commands accelerate battery cycle degradation and cause frequent direction reversals in the power conversion system. An action smoothing constraint is therefore incorporated into the Critic's Bellman target rather than directly into the reward, so that the value estimate absorbs the smoothing cost without destabilizing the policy gradient:

$$Q_{smooth}(s_h, s_h) = Q(s_h, s_h) - \lambda^{smooth}(s_h - s_{h-1})^2$$

where λ^{smooth} is the smoothing weight and s_{h-1} is the real-time dispatch in the previous interval. Combining this with the Bellman equation, the previous interval's action enters the Critic as part of the state, and the policy network implicitly favors actions close to the preceding action when maximizing Q_{smooth}. This approach is consistent with the temporal action smoothing method in [19], integrated into SAC's soft Bellman framework to avoid dimensional interference between the CVaR reward and external penalty terms.

Target network soft updates follow standard SAC exponential moving averaging: $\theta^{-} \leftarrow \tau\theta + (1-\tau)\theta^{-}$. The hierarchical structure requires the upper and lower target networks to update independently with different step sizes: the upper Critic updates after each complete trading-day trajectory, the lower Critic updates every interval, and

the smaller step size $\tau^H < \tau^L$ keeps upper-level parameters more stable. Experience replay buffers are stored independently to prevent sample imbalance from sparse upper-level action sampling. The training convergence criterion is that the average CVaR over 200 consecutive trading days improves by less than 10^{-3}, with SOC violation rate monitored simultaneously.

4 Simulation Experiment Design and Performance Evaluation of Coordinated Scheduling

The simulation experiments have three objectives: to verify that the hierarchical hybrid-action SAC achieves superior performance in both expected net revenue and risk metrics; to examine policy robustness across different forecast error quantile groups; and to analyze safe SOC management during sustained high-volatility intervals. Evaluation is performed on historical samples not used in model training, and four baseline strategies serve as comparisons.

4.1 Simulation Environment Boundary Conditions Based on Historical Load and Clearing Rules of a Provincial Grid

The simulation environment is built on the full-year 2022–2023 operating data of a provincial spot market, comprising 96-point day-ahead clearing price sequences, real-time clearing price sequences, and concurrent renewable output records — a total of 730 trading days. January 2022 through June 2023 (546 days) are used for policy training and hyperparameter tuning; July 2023 through December 2023 (184 days) are held out entirely for testing. The two periods are strictly non-overlapping in time, preventing distributional leakage.

The renewable unit is modeled as a wind-storage hybrid with installed capacity $P^{cap} = 100\,\text{MW}$. Storage parameters reflect common engineering practice: rated capacity $E^{cap} = 50\,\text{MWh}$, rated discharge power $P^{dis,max}$ = 25 MW, rated charge power $P^{ch,max}$ = 25 MW, charge/discharge efficiency $\eta^{ch} = \eta^{dis}$ = 0.92, SOC operating range [0.10, 0.90], equivalent cycle life 4,000 cycles, replacement cost amortized to c^{cyc} = 18 CNY/MWh. Error distributions are fitted with J = 5 output bins, M = 2 Gaussian mixture components, and lag-1 autocorrelation ρ = 0.61. Settlement parameters follow provincial spot rules: κ^+ = 0.9, κ^- = 1.2, tolerance band $\delta_h = 0.03P^{cap}$ (see Table 1).

Table 1. Simulation Environment and Energy Storage System Parameters

Category	Parameter	Value
Renewable Unit	Installed Capacity P^{cap}	100 MW
Renewable Unit	Training Period	Jan 2022 – Jun 2023 (546 days)
Renewable Unit	Testing Period	Jul 2023 – Dec 2023 (184 days)
Energy Storage	Rated Capacity E^{cap}	50 MWh
Energy Storage	Rated Discharge Power $P^{dis,max}$	25 MW

Category	Parameter	Value
Energy Storage	Rated Charge Power $P^{ch,max}$	25 MW
Energy Storage	Charge/Discharge Efficiency $\eta^{ch} = \eta^{dis}$	0.92
Energy Storage	SOC Operating Range	[0.10, 0.90]
Energy Storage	Equivalent Cycle Life	4,000 cycles
Energy Storage	Degradation Cost per Cycle c^{cyc}	18 CNY/MWh
Settlement Rules	Overgeneration Discount Coefficient κ^{+}	0.9
Settlement Rules	Undergeneration Premium Coefficient κ^{-}	1.2
Settlement Rules	Tolerance Band Width δ_h	0.03 P^cap (3 MW)
Error Distribution	Output Bins J	5
Error Distribution	Gaussian Mixture Components M	2
Error Distribution	Lag-1 Autocorrelation ρ	0.61

Four baseline strategies are evaluated. Strategy 1 (DA-Point) submits the day-ahead point forecast directly as the bid quantity for all intervals and performs no real-time storage dispatch, representing the simplest operational practice. For DA-Point and DA-Quant, SOC violation statistics are computed by simulating storage operation under a passive mode: storage is initialized at SOC0 and no active real-time dispatch is executed, so SOC evolves solely due to self-discharge or pre-committed schedules. The 87 violations for DA-Point thus reflect passive SOC boundary excursions rather than active policy violations, and are not directly comparable to the active SOC management of DRL strategies, which actively manage SOC. The SOC violation columns for DA-Point and DA-Quant in Table 4 are therefore labeled separately to reflect this distinction. Strategy 2 (DA-Quant) determines bid quantities using the quantile formula derived in Section 1.2 and operates storage on a rule-based heuristic that charges during the lowest-price quartile of each day and discharges during the highest-price quartile, subject to SOC bounds; this baseline isolates the contribution of distributional bidding without learned dispatch. Strategy 3 (DRL-Flat) applies standard SAC with a flattened action space that concatenates all 96 day-ahead bid quantities and the current interval's real-time dispatch into a single continuous vector, uses only expected revenue in the reward function with no CVaR term, and imposes no hierarchical structure. In this regard, DRL-Flat essentially serves as a joint ablation model, simultaneously removing both the hierarchical action space and the

adaptive CVaR mechanism. By comparing DRL-Hier with DRL-Flat, we can directly quantify the incremental value of these two core structural designs beyond a standard, competently implemented DRL baseline. The significant performance gaps across all error groups (e.g., +71.5% revenue over DA-Point and a 5.2% reduction in SOC violation rate compared to DRL-Flat in the high-error group) empirically validate the joint contribution of the hierarchical architecture and risk-sensitive reward. This baseline directly tests whether the hierarchical architecture and risk-sensitive reward provide incremental value beyond a competently implemented DRL baseline. Strategy 4 (DRL-Hier) is the proposed method. The absence of a model-based benchmark such as MILP with scenario trees or MPC is a limitation acknowledged in Section 5; such benchmarks would provide an interpretable upper bound and are deferred to future work with access to a larger computational budget. All DRL strategies are trained from the same random initialization across five independent random seeds; results in Table 3 report the mean and standard deviation across seeds, and pairwise comparisons between DRL-Hier and DRL-Flat use a paired bootstrap test with 10,000 resamples at the trading-day level (see Table 2).

Table 2. Baseline Strategy Configurations

Strategy	Name	Day-Ahead Bidding	Real-Time Storage	Algorithm	Reward
1	DA-Point	Point forecast as direct bid	No real-time dispatch	—	—
2	DA-Quant	Quantile formula (Sec. 1.2)	Rule-based strategy	—	—
3	DRL-Flat	Flattened hybrid action, standard SAC	Standard SAC continuous output	Single-level flat	Expected revenue
4	DRL-Hier	Hierarchical hybrid-action SAC	Hierarchical lower-level dispatch	Upper discrete + lower continuous	Expected revenue + adaptive CVaR + degradation cost

All DRL results in Table 3 report the mean across five independent random seeds; the standard deviation column for DRL strategies reflects cross-seed variability rather than within-strategy day-to-day variability. Pairwise differences between DRL-Hier and DRL-Flat are assessed via paired bootstrap resampling (10,000 resamples at the trading-day level): the 95% bootstrap confidence interval for the revenue difference in the high-error group is [4.1, 6.3] $\times$ 10k CNY ($p < 0.01$), and in the medium-error group is [2.4, 3.8] $\times$ 10k CNY ($p < 0.01$). Non-DRL baselines are deterministic given the price and output sequences and therefore report single-run results.

4.2 Net Revenue Comparison Across Forecast Error Quantile Groups

The test set is partitioned by realized day-ahead forecast error quantile into three groups: the low-error group (mean absolute error below the 30th percentile of the training set, 47 days), the medium-error group (30th – 70th percentile, 89 days), and the high-error group (above the 70th percentile, 47 days). All DRL results in Table 3 report the mean and standard deviation across five independent random seeds. For the

primary pairwise comparison between DRL-Hier and DRL-Flat, paired bootstrap resampling with 10,000 resamples at the trading-day level yields the following 95% confidence intervals for the revenue difference: low-error group [0.5, 1.1]×10k CNY ($p=0.003$); medium-error group [2.4, 3.8]×10k CNY ($p<0.001$); high-error group [4.1, 6.3]×10k CNY ($p<0.001$). Cohen's d effect sizes are 0.38, 0.89, and 1.24 for the three groups respectively, indicating that the advantage grows from small-to-moderate in calm conditions to large under high forecast error — the regime where the hierarchical structure and adaptive CVaR mechanism are most frequently activated. Results are shown in Table 3.

Table 3. Net Revenue, Standard Deviation, CVaR, and SOC Violation Rate by Forecast Error Group

Error Group	Strategy	Daily Avg. Net Revenue (10k CNY)	Std. Dev. (10k CNY)	$CVaR_{0.95}$ (10k CNY)	SOC Violation Rate (%)
Low Error	DA-Point	42.3	5.1	−9.8	—
Low Error	DA-Quant	43.1	4.8	−9.2	—
Low Error	DRL-Flat	44.8	4.5	−8.6	0.8
Low Error	DRL-Hier ✓	45.6	4.1	−7.9	0.4
Medium Error	DA-Point	35.7	8.4	−16.1	—
Medium Error	DA-Quant	38.2	7.6	−14.3	—
Medium Error	DRL-Flat	40.1	7.0	−12.4	4.8
Medium Error	DRL-Hier ✓	43.2	6.1	−8.7	1.6
High Error	DA-Point	21.4	12.7	−28.3	—
High Error	DA-Quant	26.8	10.9	−22.6	—
High Error	DRL-Flat	31.5	9.4	−17.8	7.3
High Error	DRL-Hier	36.7	7.8	−11.2	2.1

Error Group	Strategy	Daily Avg. Net Revenue (10k CNY)	Std. Dev. (10k CNY)	$CVaR_{0.95}$ (10k CNY)	SOC Violation Rate (%)
	r ✓				

In the low-error group, differences across strategies are modest. DA-Point reaches approximately 42.3 × 10k CNY, DA-Quant 43.1 × 10k CNY, DRL-Flat 44.8 × 10k CNY, and DRL-Hier 45.6 × 10k CNY. The small gap is because accurate forecasts keep imbalance penalties low; DRL-Hier's advantage here mainly reflects storage peak-valley price arbitrage.

In the medium-error group, strategies begin to diverge. DA-Point suffers frequent undergeneration from high bid quantities, pulling net revenue down to approximately 35.7 × 10k CNY with standard deviation expanding to 8.4 × 10k CNY. DA-Quant holds the mean at 38.2 × 10k CNY through quantile bidding but retains a high CVaR (−14.3 × 10k CNY). DRL-Flat achieves 40.1 × 10k CNY daily but with an SOC violation rate of 4.8%, near the safety threshold. DRL-Hier reaches 43.2 × 10k CNY daily, standard deviation 6.1 × 10k CNY, CVaR −8.7 × 10k CNY, and SOC violation rate 1.6%, leading on both revenue and risk dimensions.

The high-error group is the critical test of robustness. DA-Point average net revenue falls to approximately 21.4 × 10k CNY; imbalance penalties average 9.6 × 10k CNY per day, eroding nearly one-third of gross revenue. DA-Quant improves to approximately 26.8 × 10k CNY, but extreme loss events occur 8 times. DRL-Flat reaches approximately 31.5 × 10k CNY, but the SOC violation rate reaches 7.3%. DRL-Hier reaches approximately 36.7 × 10k CNY daily, standard deviation 7.8 × 10k CNY, CVaR −11.2 × 10k CNY, SOC violation rate 2.1%, with only 2 extreme loss events.

DRL-Hier's net revenue advantage over DA-Point grows monotonically with forecast error: +7.8% in the low-error group (45.6 vs. 42.3 × 10k CNY), +21.0% in the medium-error group (43.2 vs. 35.7 × 10k CNY), and +71.5% in the high-error group (36.7 vs. 21.4 × 10k CNY). Measured against DA-Quant, the corresponding improvements are +5.8%, +13.1%, and +36.9%, confirming that the advantage persists even when the non-DRL baseline incorporates distributional bidding. This pattern shows that the hierarchical structure and CVaR reward derive their value from handling uncertainty rather than exploiting mean conditions. DRL-Hier also achieves the lowest $CVaR_{0.95}$ loss and the lowest revenue standard deviation across all three groups, with the risk advantage most pronounced in the high-error group where the adaptive weight mechanism is most frequently activated.

4.3 SOC Safety Violation Statistics and Dispatch Trajectory Analysis During Sustained High-Volatility Intervals

Sustained high-volatility intervals are defined as continuous stretches where the absolute forecast error exceeds 15% of installed capacity for four or more consecutive intervals. A total of 23 such events are identified, with durations ranging from 4 to 21 intervals across 14 different trading days, totaling 273 intervals. These events are

concentrated during summer thunderstorm periods and late-autumn cold front passages, consistent with the meteorological characteristics of the province during the July–December test period.

SOC violation counts across 273 total intervals in the 23 events reveal a clear stratification. DA-Point records 87 violations (31.8%). Under the passive simulation protocol for DA-Point, no real-time storage dispatch is executed (sh=0 for all h); SOC evolves solely from self-discharge at a rate of 0.02% per interval and any pre-committed day-zero schedule. The 87 violations therefore reflect passive SOC boundary excursions: when the initial SOC is near the lower bound and self-discharge accumulates over multiple intervals, SOC drifts below SOCmin without any active compensation. DA-Quant records 63 violations (23.1%), as rule-based charging partially replenishes SOC but cannot anticipate the duration or magnitude of sustained deviation events. DRL-Flat records 41 violations (15.0%): the learned policy has acquired some SOC-protecting behavior through trial and error, but the flat action space forces a single gradient signal to simultaneously optimize 96 day-ahead bids and the current dispatch, and the absence of a CVaR term means the policy does not systematically pre-position SOC ahead of high-volatility intervals. DRL-Hier records 11 violations (4.03%), below the 5% calibration threshold. These 11 violations are not failures of constraint enforcement but are the rational outcome of the soft-penalty design: the reward function penalizes SOC excursions with coefficient $\lambda_{SOC}=500$ but does not impose a hard projection, so during the 9 intervals where absolute forecast error exceeds 22% of installed capacity, the policy accepts a bounded SOC penalty rather than forcing a storage direction reversal that would create a large positive deviation settling at the discounted $\kappa+$ rate — a net revenue loss that exceeds the SOC penalty in expectation. The remaining 2 violations arise from numerical discretization of the SOC recursion at the boundary. For deployment contexts where hard feasibility is required, a terminal projection layer that clips sh to the feasible interval defined in Section 2.1 can be appended without retraining; under stated assumptions of time-invariant market rules and SOC recursion structure, this projection preserves all coupled constraints by construction, at the cost of a small and bounded revenue reduction.

A representative high-volatility event (trading day 143 of the test set, 8 intervals, forecast error range [−18%, +21%] P^cap) is selected for trajectory analysis. On that day, the day-ahead clearing price reaches 463 CNY/MWh during the afternoon peak, and the real-time price rises to 598 CNY/MWh during 16:00–17:45 due to system stress — a day-ahead to real-time price spread of 135 CNY/MWh that significantly amplifies imbalance costs. DRL-Hier's policy pre-adjusts SOC to 0.74 before the event starts (at 15:30), up from 0.61 an hour earlier, showing that the adaptive CVaR weight triggered defensive pre-charging upon detecting rising price volatility. During the event, the policy prioritizes using storage to buffer undergeneration, smoothly drawing SOC from 0.74 to 0.38 over 8 intervals while staying within [SOC^min, SOC^max] throughout. The maximum step-to-step change in real-time dispatch magnitude is 4.2 MW, with the action smoothing constraint effectively suppressing power ramps.

DRL-Flat, under the same event, sees SOC drop to 0.08 at the 5th interval, triggering a violation. The policy is then forced to switch to charging, causing positive deviations that settle at a discount, reducing daily net revenue by approximately 7.3 × 10k CNY relative to DRL-Hier (see Fig 5). DA-Quant also falls approximately 5.8 × 10k CNY short of DRL-Hier, with most of the loss from undergeneration penalties at the 6th and 7th intervals when the $\kappa^- = 1.2$ premium coefficient substantially inflates the cost of shortfall.

Aggregating daily net revenues across all 23 high-volatility events: DRL-Hier averages 29.4 × 10k CNY, DRL-Flat 23.1 × 10k CNY, DA-Quant 19.7 × 10k CNY, and DA-Point 13.2 × 10k CNY. The timing of SOC pre-adjustment behavior (on average 2.8 intervals ahead of events) aligns with the trigger threshold of the adaptive CVaR weight, and is empirically consistent with the intended mechanism whereby rising price volatility activates the exponential weight amplification and shifts dispatch toward conservative pre-charging; causal attribution would require a controlled single-variable ablation holding all other design choices fixed (see Table 4).

Table 4. SOC Violation Count and Daily Net Revenue Comparison Across Sustained High-Volatility Events

Strategy	SOC Violations (intervals)	Violation Rate (%)	Avg. Daily Net Revenue (10k CNY)	Difference vs. DRL-Hier (10k CNY)
DA-Point	87	31.8	13.2	−16.2
DA-Quant	63	23.1	19.7	−9.7
DRL-Flat	41	15.0	23.1	−6.3
DRL-Hier ✓	11	4.03	29.4	—

Note: Violation rate = violated intervals / total intervals across 23 events (273 intervals). ✓ DRL-Hier is the benchmark. Of its 11 violations, 9 occur when absolute forecast error exceeds 22% of installed capacity.



Fig 5. Storage SOC and Real-Time Dispatch Trajectories During a Typical High-Volatility Event

Conclusion

This paper identifies and resolves three structural problems that have limited prior DRL-based approaches to joint renewable bidding and storage scheduling: implicit information leakage through poorly partitioned state spaces, flattened action spaces that destroy the causal hierarchy between day-ahead and real-time decisions, and fixed risk weights that apply the same tail-risk penalty regardless of current market stress. It is important to note that the SOC constraint management in HHA-SAC is achieved through a soft-penalty mechanism rather than a hard projection: the reward function penalizes SOC excursions with coefficient λ_{SOC} but does not enforce a hard feasibility guarantee at every interval. The 4.03% violation rate observed during sustained high-volatility events therefore reflects deliberate revenue-optimal trade-offs under extreme forecast errors, not a constraint enforcement failure. For deployment contexts where hard feasibility is required, a terminal projection layer that clips sh to the feasible interval defined in Section 2.1 can be appended without retraining; under the stated assumptions of time-invariant market rules and SOC recursion structure, this projection preserves all coupled SOC and power constraints by construction, at the cost of a bounded and quantifiable revenue reduction.

The experimental evidence on 184 out-of-sample trading days from a real provincial spot market supports three conclusions. First, the revenue advantage of HHA-SAC over non-DRL baselines is not uniform but grows monotonically with forecast error magnitude — +7.8%, +21.0%, and +71.5% over DA-Point in low-, medium-, and high-error groups — consistent with the view that the structural advantages of the framework are most valuable when uncertainty is highest, though

the absence of single-variable ablations means the contribution of each component cannot be fully disentangled. Second, the adaptive CVaR weight is associated with measurable proactive behavior: SOC is pre-positioned an average of 2.8 intervals ahead of detected high-volatility events, a pattern that does not emerge in the fixed-weight comparison; this association is empirically consistent with the intended mechanism, although causal attribution would require a controlled ablation holding all other design choices fixed. Third, the 11 violations recorded by HHA-SAC during sustained high-volatility events — yielding a 4.03% rate against 273 total intervals — are concentrated in the 9 intervals where absolute forecast error exceeds 22% of installed capacity, confirming that the soft-penalty design trades a bounded SOC excursion for a larger avoided imbalance cost in expectation under the stated assumptions.

On physical constraints, the maximum charge/discharge power, SOC recursion equations, and composite output characteristics are translated into dynamic bounds on the bidding feasible region. The composite external characteristic curve gives the interval-by-interval dynamics of the feasible region explicit representation within the reinforcement learning framework.

On algorithm architecture, a nested action space is designed with upper-level discrete day-ahead bids and lower-level continuous real-time charge/discharge decisions. The upper level uses Gumbel-Softmax relaxation for differentiable discrete policy gradients; the lower level uses reparameterization sampling for feasible continuous actions. On algorithm architecture, a nested action space is designed with upper-level discrete day-ahead bids and lower-level continuous real-time charge/discharge decisions. The upper level uses Gumbel-Softmax relaxation for differentiable discrete policy gradients; the lower level uses reparameterization sampling for feasible continuous actions. The two Critic networks estimate value independently and update asynchronously via soft updates; the non-anticipativity constraint is satisfied by construction through the information-set-partitioned state design described in Section 1.3, without requiring any penalty term in the objective.

On reward design, expected net revenue, storage degradation cost, CVaR tail penalties, and SOC violation penalties are integrated into a unified reward function. An adaptive risk weight tied to real-time market volatility keeps the policy revenue-focused in calm periods and automatically strengthens risk management in volatile periods. The action smoothing constraint is embedded in the soft Bellman target to prevent dimensional interference.

Simulation results on real data from a provincial spot market show that the proposed method outperforms all baselines on daily average net revenue, revenue standard deviation, and CVaR, with the advantage growing as forecast errors increase. During sustained high-volatility events, the SOC violation rate stays below 5%, and the policy exhibits proactive SOC pre-adjustment triggered by price volatility, validating the effectiveness of the adaptive risk weight mechanism.

Several directions remain open for future work. The current model rests on the price-taker assumption; incorporating price impact into a game-theoretic framework when storage capacity is large is a natural extension. The first-order autoregressive

copula used for temporal error correlations may not capture non-stationary structures under extreme weather; a state-dependent dynamic copula could improve environment fidelity. As power spot markets evolve toward intraday multi-round rolling clearing, extending the hierarchical architecture to three timescales — day-ahead, intraday, and real-time — is another important future direction.

While the DRL-Flat baseline provides a joint ablation of the hierarchical structure and adaptive CVaR, a limitation of this study is the absence of single-variable ablation experiments (e.g., hierarchical architecture with expected-revenue-only, or flat action space with adaptive CVaR). Such isolated ablations would help disentangle the exact marginal contribution of each component. Future work will conduct comprehensive single-variable ablation studies, along with implementing a strong model-based optimization benchmark (such as MILP with scenario trees) to provide a tighter performance upper bound.

Funding

No external funding was received for this work.

Author Contributions

Dalu Li contributed to conceptualization, methodology, software, and writing—original draft. Xiaoliang Li contributed to data curation, formal analysis, and visualization. Hanchen contributed to validation, investigation, and writing—review and editing. Lele Sun contributed to supervision and project administration. All authors read and approved the final manuscript.

Competing Interests

The authors declare no competing interests.

Data Availability

The datasets analyzed during the current study are available from the corresponding author upon reasonable request.

Ethics Approval and Consent to Participate

Not applicable. This study did not involve human participants, human data, or animals.

Consent for Publication

Not applicable.

Consent to Publish

Written informed consent to publish this manuscript has been obtained from all authors.

References

[1] Wang, Y., Zou, R., Liu, F., Zhang, L., & Liu, Q. (2021). A review of wind speed and wind power forecasting with deep neural networks. Applied Energy, 304, 117766. https://doi.org/10.1016/j.apenergy.2021.117766

[2] Yang, D., Wang, W., Gueymard, C. A., Hong, T., Kleissl, J., Huang, J., Perez, M., Perez, R., Bright, J. M., Xia, X., van der Meer, D., & Peters, I. M. (2022). A review of solar forecasting, its dependence on atmospheric sciences and implications for grid integration: Towards carbon neutrality. Renewable & Sustainable Energy Reviews, 161, 112348. https://doi.org/10.1016/j.rser.2022.112348

[3] H. Bludszuweit, J. A. Dominguez-Navarro and A. Llombart, "Statistical Analysis

of Wind Power Forecast Error," in IEEE Transactions on Power Systems, vol. 23, no. 3, pp. 983-991, Aug. 2008, doi: 10.1109/TPWRS.2008.922526.
[4] B. -M. Hodge and M. Milligan, "Wind power forecasting error distributions over multiple timescales," 2011 IEEE Power and Energy Society General Meeting, Detroit, MI, USA, 2011, pp. 1-8, doi: 10.1109/PES.2011.6039388.
[5] Coles, S. (2001). An Introduction to Statistical Modeling of Extreme Values. Springer.
[6] Pinson, P., Papaefthymiou, G., Klockl, B., Nielsen, H. A., & Madsen, H. (2009). From probabilistic forecasts to statistical scenarios of short-term wind power production. Wind Energy, 12(1), 51-62. https://doi.org/10.1002/we.284
[7] P. Pinson, C. Chevallier and G. N. Kariniotakis, "Trading Wind Generation From Short-Term Probabilistic Forecasts of Wind Power," in IEEE Transactions on Power Systems, vol. 22, no. 3, pp. 1148-1156, Aug. 2007, doi: 10.1109/TPWRS.2007.901117.
[8] J. M. Morales, A. J. Conejo and J. Pérez-Ruiz, "Short-Term Trading for a Wind Power Producer," IEEE PES General Meeting, Minneapolis, MN, USA, 2010, pp. 1-1, doi: 10.1109/PES.2010.5589379.
[9] Browell, J., & Gilbert, C. (2022). Predicting Electricity Imbalance Prices and Volumes: Capabilities and Opportunities. Energies, 15(10), 3645. https://doi.org/10.3390/en15103645
[10] Shapiro A, Dentcheva D, Ruszczyński A. Lectures on Stochastic Programming: Modeling and Theory. Philadelphia: SIAM, 2009.
[11] Birge J R, Louveaux F. Introduction to Stochastic Programming. 2nd ed. New York: Springer, 2011.
[12] Xu, B.; Zhao, J.; Zheng, T.; Litvinov, E.; Kirschen, D.S. Factoring the Cycle Aging Cost of Batteries Participating in Electricity Markets. IEEE Trans. Power Syst. 2017, 33, 2248–2259, https://doi.org/10.1109/tpwrs.2017.2733339.
[13] Dulac-Arnold, G., Evans, R., van Hasselt, H., Sunehag, P., Lillicrap, T., Hunt, J., ... & Coppin, B. (2015). Deep reinforcement learning in large discrete action spaces. arXiv:1512.07679.
[14] Kingma D P, Welling M. (2014) Auto-encoding variational Bayes. Proceedings of the 2nd International Conference on Learning Representations.
[15] Jang E, Gu S, Poole B. Categorical reparameterization with Gumbel-Softmax. Proceedings of the 5th International Conference on Learning Representations, 2017.
[16] Rockafellar R T, Uryasev S. Optimization of conditional value-at-risk. Journal of Risk, 2000, 2(3): 21-41.
[17] Cao, D., Hu, W., Xu, X., Dragicevic, T., Huang, Q., Liu, Z., Chen, Z., & Blaabjerg, F. (2020). Bidding strategy for trading wind energy and purchasing reserve of wind power producer–A DRL based approach. International Journal of Electrical Power & Energy Systems, 117, 1-10. Article 105648. https://doi.org/10.1016/j.ijepes.2019.105648
[18] Haarnoja T, Zhou A, Hartikainen K, et al. Soft actor-critic algorithms and applications. arXiv:1812.05905, 2018.
[19] Mysore, S., Mabsout, B.E., Mancuso, R., & Saenko, K. (2020). Regularizing

Action Policies for Smooth Control with Reinforcement Learning. 2021 IEEE International Conference on Robotics and Automation (ICRA), 1810-1816.