Proceedings of Machine Learning Research **318** (2026)
The 39th Canadian Conference on Artificial Intelligence

Canadian AI 2026
25–29 May 2026, Vancouver, BC

Reinforcement Learning–Based Wind Farm Layout Optimization Using Neural Surrogate Models and Real Wind Data

Ali Abbas Zaidi[†,*]
[†] University of Manitoba

Abstract

This paper presents a data-driven framework for optimizing wind farm layout that integrates a neural surrogate model with reinforcement learning to capture wake interactions while reducing the computational cost of physics-based simulations. The surrogate is trained on Reynolds-Averaged Navier–Stokes (RANS) data to predict turbine power from local flow features. Validation in a closed test set shows normalized prediction errors below 5%.

Keywords: Reinforcement Learning, Neural Surrogate Models, Simulation-Based Optimization, Renewable Energy Systems.

1. Introduction

Wind energy has emerged as one of the fastest-growing renewable resources, offering a direct pathway to reducing fossil fuel dependence and mitigating greenhouse gas emissions [1]. However, the prevailing model of large-scale wind farms sited in rural or remote regions — adopted to limit environmental impact — imposes substantial transmission losses, infrastructure costs, and diminished grid resilience [2]. These limitations have motivated a shift toward distributed and decentralized wind energy systems, especially in urban and peri-urban settings where local generation can improve both efficiency and supply reliability [2, 3].

Vertical axis wind turbines (VAWTs) are particularly well suited to such decentralized deployment. Unlike horizontal axis wind turbines (HAWTs), which require consistent inflow and yaw alignment, VAWTs are omni-directional and remain effective under the turbulent, multi directional wind conditions characteristic of built environments [4]. Their compact footprint, reduced noise emissions, and simplified maintenance further enable denser placement and higher power density in spatially constrained sites [4, 5], making them a compelling option for micro-wind and smart-city applications.

Despite these advantages, optimizing dense VAWT layouts remains an open challenge. Current industry practice relies largely on empirical spacing rules and evolutionary optimization methods — such as genetic algorithms and particle swarm optimization — coupled with simplified analytical wake models [6, 7]. While adequate for conventional HAWT farms, these approaches struggle in dense VAWT deployments, where sub-diameter spacing and strongly coupled wake interactions dominate the aerodynamic response. High-fidelity CFD simulations can capture these effects efficiently, but their computational cost renders them impractical for large-scale layout optimization [6, 7], imposing a persistent trade-off between physical accuracy and computational tractability.

To address this limitation, the current paper proposes a data-driven framework that integrates neural network surrogate modeling with reinforcement learning for urban VAWT layout optimization. CFD simulations are used to train fast surrogate models that predict turbine power output, which are then embedded within an RL environment to learn optimal placement policies under realistic wind variability derived from historical data for

[*] syed.zaidi@umanitoba.ca

Winnipeg, Manitoba. By decoupling high-fidelity flow modeling from iterative optimization, the proposed approach bridges the gap between physical realism and scalable design.

2. Related Work

Wind farm layout optimization has long been approached through heuristic and evolutionary algorithms — Genetic Algorithms (GA), Particle Swarm Optimization (PSO), Simulated Annealing (SA), Ant Colony Optimization (ACO), and Differential Evolution (DE) — which effectively explore nonlinear design spaces [8–10]. While capable for static problems, such methods demand extensive parameter tuning and repeated calls to expensive flow solvers, limiting their scalability under time-varying wind conditions [10, 11].

To reduce this cost, surrogate modeling techniques such as Kriging, response surface methods, and Bayesian optimization have been embedded within optimization workflows [4, 11–13]. These surrogates alleviate simulation expense but often degrade in high-dimensional spaces or under strongly nonlinear wake interactions [14, 15]. More recently, neural networks trained on CFD data have emerged as powerful alternatives, delivering fast inference while preserving key physical trends in wake modeling and power prediction [16–18].

In parallel, reinforcement learning (RL) has drawn growing interest in engineering optimization by learning adaptive policies through environment interaction rather than solving fixed objectives [19, 20]. Deep RL methods such as DQN and PPO have achieved strong performance on complex, non-convex energy and flow problems [21, 22], and have recently been applied to wind farm layout optimization — typically with simplified wake models [23–26]. Most existing approaches do not integrate high-fidelity CFD data directly into the learning loop, limiting physical realism. This work addresses this gap by combining CFD-trained neural surrogates with RL in a unified framework for VAWT layout optimization under real wind data, enabling scalable, physically grounded optimization.

3. Problem Formulation

3.1. Methodology

Figure 1 illustrates the proposed surrogate-assisted reinforcement learning framework for VAWT wind farm layout optimization, structured in two stages. The offline stage generates RANS CFD wake interaction data to train a deep neural network surrogate and clusters ten years of Winnipeg wind records into 12 representative scenarios via a GMM and K-means pipeline. The online stage deploys a reinforcement learning agent that sequentially places turbines, evaluating candidate layouts using fast surrogate-based power predictions under stochastically sampled wind conditions. This two-stage architecture enables broad layout exploration while preserving CFD-resolved wake physics.

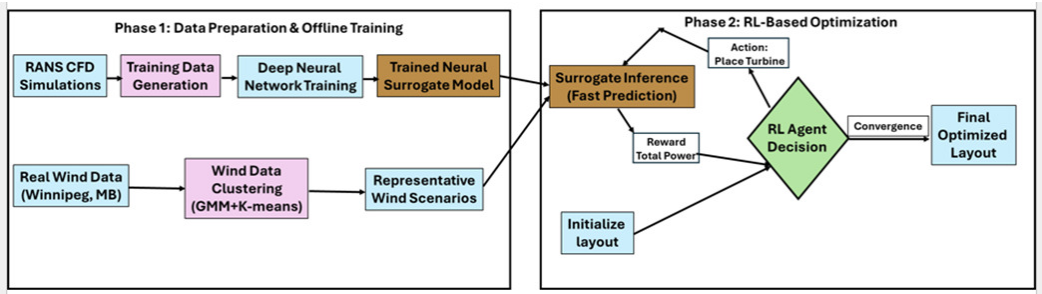

Figure 1. Surrogate-assisted reinforcement learning framework for VAWT wind farm layout optimization.

3.2. **Selection of Wind Turbine**

The ATO 400 W VAWT [27] is selected as the reference turbine for this study. A key advantage of VAWTs for this application is their omnidirectional operation—they do not require yaw alignment and maintain consistent performance regardless of wind direction. This reduces sensitivity to the multidirectional, turbulent wind conditions typical of urban environments. The turbine is designed for low wind speeds and has a cut-in speed of 2 m/s, making it suitable for the wind characteristics of Winnipeg. The specifications are summarized in Table 1. While turbine selection is not optimized, fixing the turbine type allows the study to focus on layout and density optimization, which remains highly non-trivial due to wake-induced power losses.

Turbine Feature	**Value**
Rated power	400 W
Rated wind speed	12 m/s
Cut-in wind speed	2 m/s
Rotor diameter	0.62 m
Blade height	0.9 m
Hub height	1.5 m

Table 1. Turbine specifications.

All simulations are conducted for a wind farm site in Winnipeg, Canada (49°54′36″ N, 97°14′24″ W; elevation 238.70 m). A fixed rectangular domain is considered, with properties summarized in Table 2. The farm is oriented such that the positive X-direction points south. A grid-based layout is adopted, with turbines arranged uniformly at a fixed spacing of one rotor diameter (1.0D = 0.62 m) in both streamwise and cross-stream directions, yielding a 100 × 100 grid within a 100D × 100D (62 m × 62 m) domain. Here, 1.0D defines the minimum allowable spacing and is independent of the 1.5D radius used for surrogate feature extraction. Although 1.0D represents an aggressive configuration, prior studies on closely spaced VAWT arrays [6] show that dense layouts can achieve high power density. The grid resolution can be adjusted (e.g., 1.5D or 2.0D) without modifying the surrogate or reinforcement learning framework.

Farm property	**Value**
Location	Winnipeg, Manitoba
Land size (length × width)	$100D \times 100D$
Orientation	North to South lengthwise
Density of air	1.2 kg/m^3

Table 2. Wind farm site and environmental parameters used in the layout optimization study.

3.3. **Wind Conditions and Data Representation**

The site's wind resource is characterized using ten years of hourly data from Environment Canada [28], collected at the Winnipeg Richardson International Airport weather station (Station ID: 3698), located within 1 km of the site and measured at a 10 m reference height. Wind speeds are extrapolated to the turbine hub height using a logarithmic wind profile.

$$\frac{U}{U_m} = \frac{\ln(z/z_0)}{\ln(z_m/z_0)} \tag{3.1}$$

where U_m is the measured wind speed at height z_m, z is the turbine hub height, and $z_0 = 0.1$ m represents the surface roughness length for open terrain. Each wind observation is represented by its speed V and direction θ. To enable efficient learning and avoid directional discontinuities, wind data are mapped into Cartesian coordinates,

$$u = V\cos\theta, \quad v = V\sin\theta \tag{3.2}$$

To reduce computational cost during layout optimization using reinforcement learning, the full wind dataset is condensed into a set of 12 representative clusters (as shown in Table 3) using Gaussian Mixture Model (GMM) clustering with Bayesian Information Criterion (BIC)-based model selection [29, 30], followed by k-means refinement [31]. Each resulting cluster represents a statistically significant wind condition and is associated with a frequency weight reflecting its occurrence probability.

Cluster	Speed (m/s)	Dir. Mean (°)	Dir. Std (°)	Speed Std (m/s)	% of Data	Density Rank
1	6.88	65.4	102.5	1.41	7.4	11
2	3.06	169.5	18.0	1.02	10.5	3
3	2.97	296.7	18.6	1.01	12.5	2
4	2.96	70.2	23.6	1.14	8.0	8
5	6.15	186.0	14.9	1.01	10.9	1
6	2.97	175.1	165.7	1.13	8.5	4
7	6.01	329.4	14.4	1.12	7.8	6
8	6.32	271.9	18.6	1.26	7.3	7
9	3.18	236.8	17.5	1.11	9.3	5
10	10.46	294.1	94.4	1.74	5.5	12
11	9.81	174.5	18.0	1.51	6.0	9
12	5.88	125.8	19.6	1.40	6.4	10

Table 3. Representative wind clusters derived from historical wind data.

3.4. **Training Data Generation and Network Architecture**

Training data for the surrogate model are generated using Reynolds-Averaged Navier–Stokes (RANS) CFD simulations across a range of inter-turbine spacings and inflow conditions (Table 4). For each layout, the flow field around individual turbines is sampled at predefined locations within a 1.5 rotor-diameter (1.5D) neighborhood, capturing both upstream inflow and near-wake interactions. This 1.5D radius defines only the local sampling region for feature extraction and is independent of the inter-turbine spacing constraint used in layout optimization, which remains fixed at 1.0D as specified in Section 3.2. The sampled velocity values form the input feature vector to the surrogate model. Corresponding turbine power output is obtained directly from the CFD simulations by computing the aerodynamic torque on the blades and multiplying it by the turbine's angular velocity as given in Eq.3.3.

$$P = T\,\omega \tag{3.3}$$

Wind speed (m/s)	2, 4, 6, 10, 12
Streamwise spacing (L_x/D)	2.0–10.0 (change = $0.5D$) 11.0–15.0 (change = $1.0D$)
Lateral spacing (L_y/D)	1.0, 1.5, 2.0, 2.5

Table 4. Input parameter ranges for the CFD-based surrogate training campaign, encompassing wind speeds from 2 to 12 m/s and streamwise spacings from $2.0D$ to $15.0D$ at four lateral separation distances.

where T denotes the blade-resolved torque and ω is the turbine rotational speed. The surrogate is implemented as a fully connected feedforward neural network that maps local flow features to turbine power output. The input consists of 48 normalized velocity samples per turbine, extracted from a 1.5D neighborhood around the rotor. The network comprises five hidden layers with widths 128–256–256–128–64, each using ReLU activation to capture nonlinear wake–power relationships. Batch normalization is applied after each hidden layer to stabilize training and accelerate convergence, and dropout (rate = 0.2) is used in the first three layers to improve generalization. The output layer contains a single neuron with linear activation, yielding the predicted turbine power (W). The model has approximately 87,000 trainable parameters, providing sufficient expressiveness while maintaining low inference cost.

The training dataset comprises approximately 500 turbine-level samples generated from high-fidelity CFD simulations covering diverse turbine configurations, spacings, and inflow conditions (Table 4). The data are split into training (70%), validation (15%), and test (15%) sets using stratified sampling to preserve wind-condition distributions. The model is trained by minimizing mean squared error (MSE) using the Adam optimizer with an initial learning rate of 0.001 and exponential decay. Training proceeds for up to 200 epochs, with early stopping based on validation loss; convergence is typically achieved within $\sim$85 epochs.

Although the dataset size is modest by deep learning standards, alternative models commonly recommended for small tabular datasets—Random Forest, XGBoost, and Support Vector Regression—are also evaluated. Using identical splits, tree-based methods achieve competitive performance ($R^2 = 0.91$–0.92), while the neural surrogate provides the best accuracy ($R^2 = 0.94$, RMSE = 19.3 W) and fastest inference (0.8 ms per sample). Although the improvement ($\sim$10–15% lower error) is moderate, it becomes significant when accumulated over $\sim$16,000 layout evaluations during RL training. The surrogate's effectiveness at this scale is supported by high-quality CFD data, moderate input dimensionality (48 features), a compact regularized architecture, and strong held-out generalization (NRMSE $<$ 5%). These results justify its selection within the proposed surrogate-assisted RL framework.

3.5. **Integration of the Surrogate Model into Layout Optimization**

Once trained and validated, the neural surrogate serves as the core computational engine of the reinforcement learning (RL)–based wind farm layout optimization framework. It replaces high-fidelity CFD simulations within the RL environment, enabling rapid evaluation of turbine placement decisions while retaining wake-induced power loss effects learned from physics-based data.

At each decision step, the RL agent proposes a turbine placement, after which the surrogate predicts both the incremental power of the new turbine and the updated outputs of existing turbines affected by wake interactions. These turbine-level predictions are aggregated to compute total farm power, which defines the reward signal. With an inference time of approximately 0.8 ms per turbine, the environment supports on the order of 10^3–10^4 layout evaluations per second, compared to 45–60 minutes per CFD evaluation, yielding a speedup exceeding four orders of magnitude.

The surrogate-enabled RL framework allows the agent to complete 400 training episodes, each comprising approximately 30–45 sequential placement decisions, resulting in more than 16,000 environment interactions. An equivalent CFD-based optimization would require over 10^4 GPU-hours, whereas the surrogate-accelerated training completes in approximately 2 hours on a single GPU.

4. Results and Discussions

4.1. Evaluation of the Surrogate Model

For validation against experimental turbine behavior, the reference power for the test set is computed using a turbine-specific power curve derived from manufacturer data and modeled as a fifth-order polynomial.

$$P(U) = 0.0264\,U^5 - 1.0275\,U^4 + 14.025\,U^3 - 78.227\,U^2 + 193.46\,U - 166.14 \tag{4.1}$$

where U is the local wind speed at the turbine hub height and $P(U)$ is the corresponding power output in watts. To ensure physical realism, the power curve is implemented in a piecewise continuous form subject to operational constraints,

$$P(U) = \begin{cases} 0 & U < U_{\text{in}} \\ P(U) & U_{\text{in}} \le U < U_r \\ P_r & U_r \le U \le U_{\text{out}} \\ 0 & U > U_{\text{out}} \end{cases} \tag{4.2}$$

where $U_{\text{in}} = 2$ m/s, $U_r = 12$ m/s, $U_{\text{out}} = 15$ m/s, and $P_r = 400$ W. To assess predictive accuracy and generalization, Table 3 reports MAE, RMSE, NRMSE, MAPE, and R^2 values computed on the test set against both the manufacturer-based power curve and RANS CFD–derived power estimates. This dual comparison evaluates the surrogate's consistency with experimental turbine behavior as well as its ability to reproduce physics-based CFD trends.

Evaluation Method	**MAE (W)**	**RMSE (W)**	**NRMSE**	**MAPE**	R^2
Manufacturer power curve	14.2	16.8	4.2%	4.1%	0.95
RANS CFD simulations	16.1	19.3	4.8%	4.9%	0.94

Table 5. Surrogate model predictive accuracy on test set.

Evaluation Method	**MAE (W)**	**RMSE (W)**	R^2	**MAPE**
Single 70–15–15 split	16.1	19.3	0.94	4.9%
5-fold cross-validation	15.1 ± 1.2	18.5 ± 1.6	0.941 ± 0.019	4.6 ± 0.4%

Table 6. Surrogate model performance using 5-fold cross-validation.

The results in Table 5 demonstrate that the surrogate achieves strong predictive accuracy and generalization on the held-out test set. When benchmarked against the manufacturer-derived power curve, it exhibits low absolute and relative errors, with normalized errors around 4% and a high coefficient of determination ($R^2 = 0.95$), indicating close agreement with experimental turbine performance. Validation against RANS CFD–based power shows slightly higher errors, as expected due to numerical variability and wake-induced fluctuations; however, the normalized error remains below 5% and R^2 exceeds 0.94, confirming robust generalization to unseen flow conditions. The consistent error levels across both references indicate that the surrogate effectively captures the dominant relationship between local flow features and turbine power output. Moreover, the prediction error (NRMSE $\approx$ 4–5%) is small relative to the turbine rated power (400 W) and significantly lower than the performance gains between optimized layouts ($\approx$ 9–10% over GA), suggesting that surrogate uncertainty does not affect layout ranking and is suitable for RL-based optimization.

Table 6 compares performance from a single 70–15–15 split with 5-fold stratified cross-validation using RANS CFD power as reference. The cross-validation metrics (RMSE =

18.5 ± 1.6 W, $R^2 = 0.941 \pm 0.019$, MAPE $= 4.6 \pm 0.4\%$) align closely with the single-split results (RMSE = 19.3 W, $R^2 = 0.94$), indicating that the reported accuracy is not sensitive to data partitioning. The low variance across folds further confirms stable generalization across CFD-generated samples, providing statistical confidence in the surrogate's predictive reliability.

4.2. **Influence of Surrogate Fidelity on Policy Learning**

In the proposed framework, the reinforcement learning (RL) agent relies exclusively on surrogate-based power predictions to compute rewards; thus, surrogate fidelity directly shapes the reward landscape during policy learning. Large or systematic prediction errors could bias the agent toward layouts that are optimal under the surrogate but suboptimal under true CFD-based physics.

In this study, the surrogate accuracy (NRMSE $\approx$ 4–5%, $R^2 \approx 0.94$) is substantially smaller than the performance gap between GA and RL layouts ($\sim$9–10% improvement in turbine density and total power). This margin suggests that surrogate uncertainty is unlikely to alter layout ranking. In addition, cross-validation shows low variability across folds (R^2 std ≈ 0.02), indicating stable reward estimation.

These results imply that surrogate noise does not dominate policy learning, and the RL agent captures meaningful physical trends rather than surrogate artifacts. More broadly, this highlights a key principle for surrogate-assisted RL: prediction error should remain smaller than the performance gap between competing design configurations.

4.3. **Optimization of the Layout of Wind Farm using Reinforcement Learning**

Once trained and validated, the neural surrogate serves as the core computational engine of the reinforcement learning (RL)–based wind farm layout optimization framework. At each decision step, the RL agent proposes a turbine placement, after which the surrogate predicts the incremental power of the new turbine and the updated outputs of existing turbines affected by wake interactions. These predictions are aggregated to compute total farm power, which defines the reward signal. With an inference time of approximately 0.8 ms per turbine, the environment supports on the order of 10^3–10^4 layout evaluations per second, compared to 45–60 minutes per CFD evaluation, yielding a speedup exceeding four orders of magnitude and enabling computationally feasible learning-based optimization.

Figure 2 shows the training progression of the RL agent, with episode-wise rewards and a 50-episode moving average. Early-stage variability reflects exploration and occasional suboptimal placements. The moving average stabilizes after approximately 150–200 episodes, indicating convergence toward a consistent policy. Occasional low-reward episodes persist due to continued exploration and stochastic wind sampling; however, the absence of sustained degradation confirms policy robustness. Overall, the stabilized moving-average reward and bounded variance indicate that the agent effectively balances turbine density and wake-induced losses, converging to a reliable optimization strategy for layout evaluation.

To evaluate learning stability and reproducibility, five independent RL agents were trained using different random seeds for environment initialization, exploration, and stochastic wind sampling. Table 7 summarizes the resulting performance statistics. The low coefficient of variation (CV) in turbine count (4.2%) and total power (3.8%) indicates consistent convergence to high-quality layouts despite stochastic. All policies outperform the genetic algorithm baseline (41 turbines, 4.3 kW), confirming the robustness of the surrogate-accelerated RL framework. The higher variability in convergence episode (CV = 23.1%) reflects differences in early exploration and does not affect final layout quality.

To further validate the proposed reinforcement learning (RL)–based framework, a genetic algorithm (GA) was implemented as a baseline under identical spatial constraints and

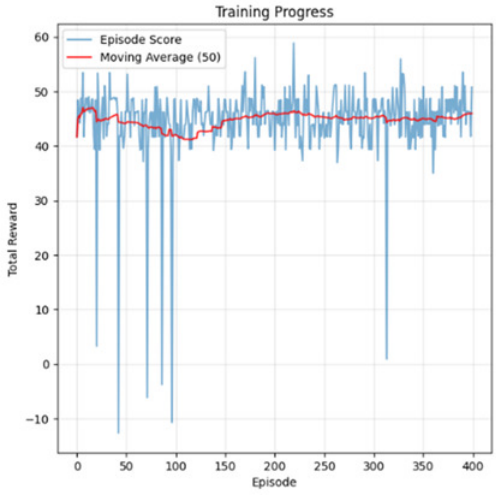

Figure 2. Reinforcement learning training history showing episode reward and 50-episode moving average.

Metric	**Mean ± Std**	**Range**	**CV (%)**
Turbines placed	44.6 ± 0.9	[44, 46]	4.2
Total power (kW)	4.72 ± 0.18	[4.51, 4.95]	3.8
Farm efficiency (%)	26.4 ± 1.2	[24.8, 27.9]	4.5
Power density (W/m^2)	47.6 ± 1.8	[45.2, 49.8]	3.8
Convergence episode	78 ± 18	[58, 102]	23.1

Table 7. RL training stability across 5 independent runs.

turbine specifications. The GA uses binary encoding over the discrete grid, with hyperparameters (population size, mutation rate, crossover probability) tuned to ensure convergence within a computational budget comparable to RL training (∼16,000 layout evaluations). Increasing generations or population size beyond this threshold yields negligible improvement (<1%), indicating practical convergence.

The GA converges to a layout with 41 turbines, representing the maximum configuration identified by the evolutionary search. In contrast, the RL-based approach (Figure 3) accommodates 45 turbines within the same domain while achieving higher power density and efficiency. This difference highlights a key limitation of GA-based optimization in high-dimensional, wake-coupled problems, where population-based heuristics may converge prematurely and struggle to exploit fine-scale spatial trade-offs.

By contrast, the RL agent leverages sequential decision-making and surrogate-based feedback to identify placements that maintain favorable wake interactions, enabling higher turbine density without disproportionate losses. Consequently, RL-optimized layouts achieve both higher turbine count (Figure (Figure 4) and improved farm-level performance, demonstrating superior scalability and adaptability. These results confirm that learning-based optimization is a more effective strategy for dense wind farm design in constrained environments.

4.4. **Comparison of the layout optimization of Wind Farm**

Table 8 positions the proposed framework relative to recent wind farm layout optimization studies in terms of modeling fidelity, optimization strategy, scalability, and computational

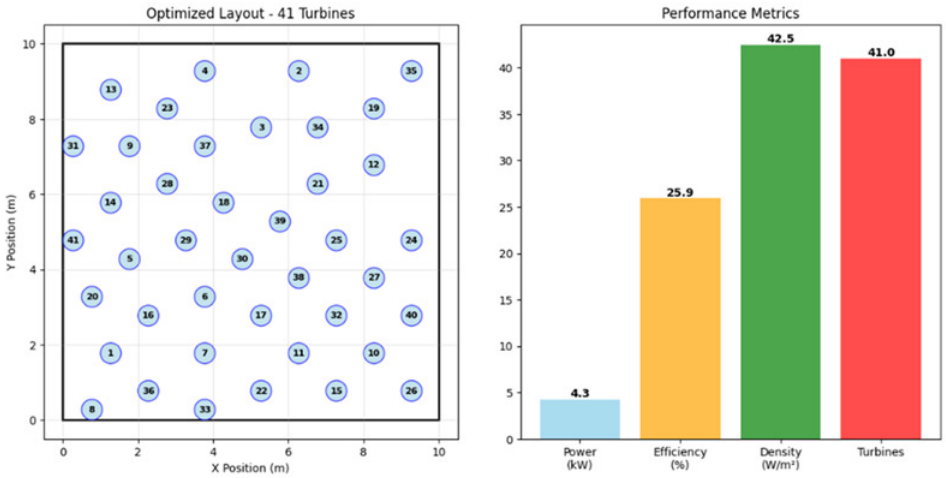

Figure 3. Wind farm layout optimized using a Genetic Algorithm (GA). The left panel shows the final turbine arrangement, resulting in 41 turbines. The right panel presents corresponding performance metrics.

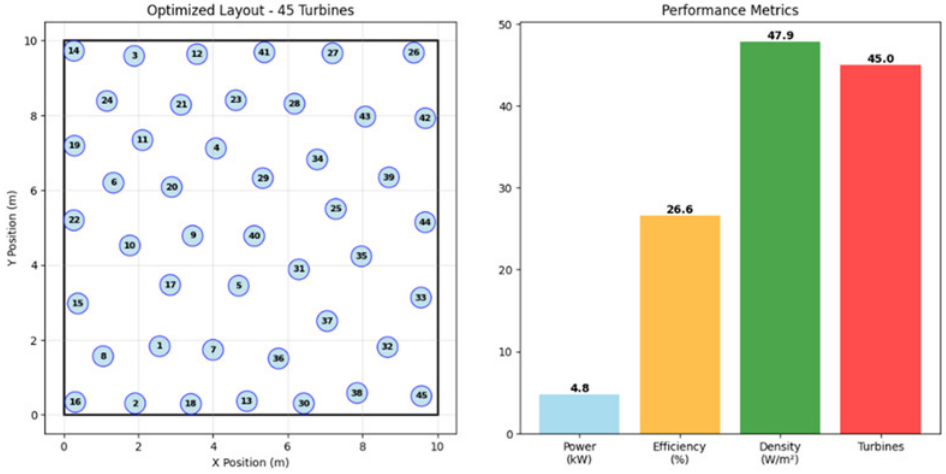

Figure 4. Wind farm layout optimized using reinforcement learning (RL). The left panel shows the final turbine arrangement, resulting in 45 turbines. The right panel presents corresponding performance metrics.

cost. Compared with analytical wake-based methods [12], which rely on simplified Jensen-type models, the present approach retains CFD-resolved wake interactions through a RANS-trained neural surrogate, achieving higher physical fidelity at practical cost.

Relative to reinforcement learning approaches based on reduced-order wake models [13, 15], the proposed framework incorporates CFD-validated wake physics without the expense of inline simulation. The surrogate achieves NRMSE $< 5\%$ ($R^2 = 0.94$), comparable to recent neural wake models [26, 27], while enabling evaluations in approximately 0.8 ms per configuration.

High-fidelity optimization using LES or full CFD [21, 28] provides detailed flow resolution but typically requires days to weeks of computation for small arrays (6–9 turbines). In

contrast, the surrogate-accelerated framework enables RL training over 400 episodes in approximately 2 hours, making large-scale layout exploration feasible.

Under identical evaluation conditions (same surrogate, wind scenarios, and constraints), the proposed RL framework achieves a 9.8% increase in turbine density relative to a genetic algorithm baseline. This gain reflects improved exploration of combinatorial placement configurations rather than differences in physical modeling. Overall, the method offers a balanced trade-off between physical fidelity, computational scalability, and optimization performance, particularly for dense VAWT deployments in constrained domains.

Study	**Turbine Type**	**Optimization Method**	**Wake Model**	**Farm Size**	**Key Performance**	**Computational Cost**
Mosetti et al. [10]	HAWT	Genetic Algorithm	Jensen analytical	2×2 km, 30 turbines	98.7% efficiency	Not reported
Vyshnav et al. [24]	HAWT	Deep Q-Network (DQN)	Simplified wake	5×5 grid, variable count	15% improvement vs. GA	Hours (not specified)
Dong et al. [22]	HAWT	PPO + LES	High-fidelity LES	6-turbine array	12% power gain	Days (LES inline)
Park and Park [21]	HAWT	Graph Neural Network	Physics-informed GNN	Variable layouts	$R^2 = 0.92$ wake prediction	Minutes (inference)
Li et al. [20]	HAWT	BiCNN surrogate	CFD-trained CNN	Wind farm wake modeling	RMSE = 7.2%	Seconds (inference)
Bempedelis & Magri [16]	HAWT	Bayesian Optimization + CFD	High-fidelity CFD	9 turbines	Pareto-optimal layouts	Weeks (with CFD)
Haddad et al. [8]	VAWT (Savonius)	ANN surrogate + optimization	CFD-trained ANN	Single turbine design	$R^2 = 0.96$ prediction	Not reported
This work	VAWT	RL + Neural Surrogate	RANS CFD-trained DNN	100×100 m, 45 turbines	NRMSE $< 5\%$, 9.8% more turbines vs. GA	2 hours (RL), 0.8 ms (inference)

Table 8. Comparison with recent wind farm layout optimization approaches.

5. Conclusions/Future Work

This paper presents a data-driven framework for vertical-axis wind turbine (VAWT) layout optimization that integrates high-fidelity CFD simulations, neural surrogate modeling, and reinforcement learning. The surrogate accurately predicts turbine-level power from localized flow features, achieving normalized errors below 5% when validated against manufacturer curves and RANS CFD results. Embedded within an RL environment, it enables efficient, wake-aware turbine placement under realistic wind conditions for Winnipeg, Canada. The RL agent demonstrates stable convergence and outperforms a genetic algorithm baseline under identical constraints, producing denser, higher-performing layouts without the cost of direct CFD-based optimization.

Although validated using Winnipeg wind data, the framework is inherently site-adaptive. The surrogate can be retrained with location-specific CFD and wind data, and the RL agent can learn policies tailored to local wind distributions. Owing to the surrogate's efficiency, retraining and optimization can be completed within approximately one day on a standard workstation, enabling practical deployment. Future work will extend validation to diverse regions and terrains to assess performance sensitivity and develop generalized design guidelines for scalable urban VAWT deployment.

6. Data and Code Availability

The wind data used in this study are publicly available from Environment and Climate Change Canada (Winnipeg Richardson International Airport station), and turbine specifications are obtained from manufacturer documentation. The neural network architecture, training procedure, and reinforcement learning framework are described in sufficient detail

to enable reproduction. Due to the large size of the CFD-generated datasets, raw simulation files are not included in the supplementary material but are available upon request.

References

[1] D. Nugent and B. K. Sovacool. "Assessing the lifecycle greenhouse gas emissions from solar PV and wind energy: A critical meta-survey". In: *Energy Policy* 65 (2014), pp. 229–244.

[2] I. Azpiri, J. Vandewalle, and D. Van Hertem. "Transmission losses in offshore wind farm collection and transmission systems". In: *European Wind Energy Association Annual Event*. 2015.

[3] Y. Cheng, X. Liu, and Z. Wang. "Decentralized wind energy systems as catalysts for urban resilience: A design framework". In: *Energy Science & Engineering* 13.6 (2025).

[4] M. Kumar, A. Saxena, and D. Kumar. "Vertical axis wind turbines in emerging energy applications (1979–2025): Global trends and technological gaps revealed by a bibliometric analysis and review". In: *Energies* 18.14 (2025), p. 3810.

[5] O. Dabiri. "Potential order-of-magnitude enhancement of wind farm power density via counter-rotating vertical-axis wind turbine arrays". In: *Journal of Renewable and Sustainable Energy* 3.4 (2011), p. 043104.

[6] S. Grady, M. Hussaini, and M. Abdullah. "Placement of wind turbines using genetic algorithms". In: *Renewable Energy* 30.2 (2005), pp. 259–270.

[7] P. A. Fleming, P. M. Gebraad, S. Lee, J. van Wingerden, K. Johnson, M. Churchfield, J. Michalakes, P. Spalart, and P. Moriarty. "Simulation comparison of wake mitigation control strategies for a two-turbine case". In: *Wind Energy* 18.12 (2015), pp. 2135–2143.

[8] H. Z. Haddad, M. H. Mohamed, Y. M. Shabana, and K. Elsayed. "Optimization of Savonius wind turbine with additional blades by surrogate model using artificial neural networks". In: *Energy* 270 (2023), p. 126952.

[9] D. Cardoso Netto, R. Ramirez Gustavo, and N. Manzanares Filho. "Surrogate-based design optimization of an H-Darrieus wind turbine comparing classical response surface, artificial neural networks, and Kriging". In: *Journal of Applied Fluid Mechanics* 16.4 (2023), pp. 703–716.

[10] G. Mosetti, C. Poloni, and B. Diviacco. "Optimization of wind turbine positioning in large windfarms by means of a genetic algorithm". In: *Journal of Wind Engineering and Industrial Aerodynamics* 51.1 (1994), pp. 105–116.

[11] J. F. Herbert-Acero, O. Probst, P.-E. Réthoré, G. C. Larsen, and K. K. Castillo-Villar. "A review of methodological approaches for the design and optimization of wind farms". In: *Energies* 7.11 (2014), pp. 6930–7016.

[12] T. A. Qureshi and V. Warudkar. "Wind farm layout optimization through optimal wind turbine placement using a hybrid particle swarm optimization and genetic algorithm". In: *Environmental Science and Pollution Research* 30 (2023), pp. 77436–77452.

[13] A. C. Pillai, J. Chick, L. Johanning, M. Khorasanchi, and S. Barbouchi. "Offshore wind farm layout optimization using particle swarm optimization". In: *Journal of Ocean Engineering and Marine Energy* 4.1 (2018), pp. 73–88.

[14] N. Dimitrov, M. C. Kelly, A. Vignaroli, and J. Berg. "From wind to loads: Wind turbine site-specific load estimation with surrogate models trained on high-fidelity load databases". In: *Wind Energy Science* 3.2 (2018), pp. 767–790.

[15] Y. Zha, M. Zhang, Z. Wan, and X. Guo. "An optimization framework for wind farm layout design using CFD-based Kriging model". In: *Ocean Engineering* 295 (2024), p. 116923.

[16] N. Bempedelis and L. Magri. "Bayesian optimization of the layout of wind farms with a high-fidelity surrogate model". In: *Computational Science – ICCS 2023, Lecture Notes in Computer Science*. Vol. 13976. Springer, Cham, 2023, pp. 333–341.

[17] C. K. J. Hou and K. Behdinan. "Dimensionality reduction in surrogate modeling: A review of combined methods". In: *Data Science and Engineering* 7 (2022), pp. 402–427.

[18] C. Sun, Y. Jin, R. Cheng, J. Ding, and J. Zeng. "Variable surrogate model-based particle swarm optimization for high-dimensional expensive problems". In: *Complex & Intelligent Systems* 9 (2023), pp. 2833–2854.

[19] Z. Li, X. Zhang, and M. D. Piggott. "Graph neural network-based wind turbine wake modeling and optimization". In: *Journal of Ocean Engineering and Marine Energy* 8 (2022), pp. 113–129.

[20] K. Li, W. Zhang, J. Liu, and X. Chen. "Dynamic wind farm wake modeling based on a bilateral convolutional neural network and high-fidelity LES data". In: *Energy* 258 (2022), p. 124845.

[21] H. Park and R. T. Park. "Physics-induced graph neural network: An application to wind-farm power estimation". In: *Energy* 187 (2019), p. 115883.

[22] H. Dong, J. Zhang, and X. Zhao. "Wind farm control technologies: From classical control to reinforcement learning". In: *Progress in Energy* 4.3 (2022), p. 032006.

[23] H. Dong, J. Zhang, and X. Zhao. "Intelligent wind farm control via deep reinforcement learning and high-fidelity simulations". In: *Applied Energy* 292 (2021), p. 116928.

[24] T. C. Vyshnav, K. C. S. Thampatty, S. Sreekumar, and K. G. Satheesh. "Reinforcement learning based wind farm layout optimization". In: *2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS)*. IEEE, 2022, pp. 612–617.

[25] M. Ghazvini, M. Gholami, S. Paudyal, and H. Wang. "Deep reinforcement learning for resilient power and energy systems: Progress, prospects, and future avenues". In: *Energies* 17.22 (2024), p. 5727.

[26] M. C. F. Bauweraerts and J. Meyers. "Reinforcement learning for wind-farm flow control: Current state and future actions". In: *Theoretical and Applied Mechanics Letters* 14.1 (2024), p. 100503.

[27] ATO. *400W vertical axis wind turbine, 12V/24V – Technical specifications*. https://www.ato.com/400w-vertical-axis-wind-turbine. Accessed: Dec. 02, 2025. 2025.

[28] Environment and Climate Change Canada. *Historical climate data*. https://climate.weather.gc.ca/historical_data/search_historic_data_e.html. Accessed: Dec. 02, 2024. 2024.

[29] C. M. Bishop. *Pattern Recognition and Machine Learning*. New York, NY: Springer, 2006.

[30] G. Schwarz. "Estimating the dimension of a model". In: *Annals of Statistics* 6.2 (1978), pp. 461–464.

[31] J. MacQueen. "Some methods for classification and analysis of multivariate observations". In: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*. Vol. 1. Berkeley, CA: University of California Press, 1967, pp. 281–297.