An Implicitly Differentiable Framework for Physics-Informed Wind Turbine Blade Design Optimization

Lucas Costa Barbosa[0009−0002−7464−6800], Adriano Mauricio de Almeida Cortes[0000−0002−2941−2522], and Laura Silvia Bahiense da Silva Leite[0000−0001−8175−8320]

Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil
{lbarbosa, adricortes, laura}@cos.ufrj.br

Abstract. The aerodynamic optimization of wind turbine blades is traditionally bottlenecked by computationally expensive, non-differentiable high-fidelity simulations that preclude efficient gradient-based design. This work presents a novel differentiable computational framework that integrates deep learning surrogates with physical solvers to enable end-to-end gradient-based optimization of rotor performance. We propose a hybrid architecture where NeuralFoil (a neural network predicting airfoil coefficients) is embedded within a GPU-accelerated, differentiable Blade Element Momentum Theory (BEMT) solver with guaranteed convergence.
Unlike standard iterative solvers, our implementation leverages the Implicit Function Theorem (IFT) to compute exact gradients of the power coefficient (C_P) with respect to high-dimensional geometric parameters, achieving numerical stability even in low-speed regimes where traditional tools like QBlade often diverge. We demonstrate the efficacy of this framework on the NREL 5MW reference turbine, using an Augmented Lagrangian method to satisfy geometric and aerodynamic constraints rigorously. The proposed pipeline runs fully in parallel on GPUs, offering a scalable, physics-informed pathway for next-generation renewable energy engineering.

Keywords: Wind Energy · Differentiable Design · GPU-Accelerated · Physics-Informed · Scientific Machine Learning.

1 Introduction

The rapid expansion of wind energy has made aerodynamic efficiency a central driver of turbine performance and economic viability. The power coefficient C_P directly determines the annual energy production of a turbine, and even modest improvements can translate into significant gains over its operational lifetime. Designing optimal blade geometries, however, remains a challenging inverse problem due to its high-dimensional design space, nonlinear aerodynamics, and costly performance evaluations.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_4

Current optimization strategies face fundamental limitations. High-fidelity approaches such as Computational Fluid Dynamics (CFD) provide accurate physical modeling but are computationally prohibitive for iterative design and large-scale optimization. In contrast, low-fidelity engineering tools based on Blade Element Momentum Theory (BEMT) are computationally efficient, yet standard implementations rely on fixed-point solvers, tabulated airfoil polars at discrete Reynolds numbers, and non-smooth control flow. As a result, these solvers are effectively non-differentiable and incompatible with modern automatic differentiation frameworks, forcing practitioners to adopt gradient-free optimization methods with poor sample efficiency in high-dimensional settings.

Despite recent advances in surrogate modeling and scientific machine learning, a gap remains between physically consistent turbine models and end-to-end differentiable optimization pipelines. Purely data-driven surrogates often struggle to enforce physical constraints or generalize beyond their training regime, while classical physics-based solvers are tightly coupled to lookup tables and iterative schemes that hinder stable gradient computation.

To address these challenges, this work introduces an implicitly differentiable, physics-informed framework for wind turbine blade design. We combine two differentiable reduced-order models within a Scientific Machine Learning (SciML) architecture. NeuralFoil [14] replaces discrete airfoil polar tables with a differentiable neural surrogate that maps airfoil shape parameters and Reynolds number to aerodynamic coefficients. This model is coupled to a GPU-accelerated BEMT solver based on the single-residual formulation of [12], which exhibits robust convergence across operating conditions.

Crucially, gradients of turbine performance with respect to design variables are computed via the Implicit Function Theorem, following the OptNet paradigm [1]. By differentiating the equilibrium residual rather than unrolling solver iterations, the proposed approach decouples memory cost from iteration count and ensures numerical stability, even in low-wind-speed regimes where conventional solvers often fail. The resulting pipeline enables efficient gradient-based optimization of blade geometry, including airfoil shape, and substantially reduces computational cost compared to gradient-free baselines, moving toward physically consistent and differentiable digital twins for aerodynamic design optimization.

2 Related Work

The optimization of aerodynamic shapes traditionally splits between high-fidelity, gradient-based methods and low-fidelity, derivative-free approaches. In Computational Fluid Dynamics (CFD), the adjoint method [7] is the gold standard, computing gradients independent of design variable count. However, discrete adjoint solvers are mathematically rigid and memory-intensive for complex multiphysics [6]. Conversely, standard low-fidelity tools like Blade Element Momentum Theory (BEMT) solvers (e.g., QBlade [11]) are computationally lightweight but rely on discrete look-up tables and fixed-point iterations. This obscures gra-

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_4

dient flow, forcing reliance on sample-inefficient metaheuristics like Genetic Algorithms [3] or noisy Finite Difference approximations [12].

Recent advances in Scientific Machine Learning (SciML) and differentiable programming (e.g., JAX-CFD [8]) offer a middle ground by embedding physical constraints into learning pipelines. While purely data-driven models [10] often fail to generalize beyond training limits, our hybrid approach restricts the neural surrogate to the airfoil level [14] while governing global performance via BEMT physics. To overcome the memory bottleneck of unrolling iterative solvers ("exploding tape"), we adopt the Deep Equilibrium paradigm [1]. By differentiating the equilibrium state via the Implicit Function Theorem (IFT), our framework calculates exact BEMT gradients without the memory overhead of standard automatic differentiation.

3 Proposed Framework and Numerical Experiments

3.1 Parametrization

The proposed framework is parametrized by two tensors. The **design tensor** $\mathbf{D} \in \mathbb{R}^{B\times S\times 24}$ stores the variables for each of the B blades and S sections, encompassing the Kulfan class exponents (N_1, N_2), 16 upper and lower Bernstein coefficients, leading/trailing edge modifications, radial position r, chord length c, twist angle θ, and cylindrical drag coefficients. The **operation tensor** $\mathbf{O} \in \mathbb{R}^{T\times S\times 9}$ defines the environmental and turbine conditions. For this study, we match the standard NREL 5MW specifications (e.g., diameter $D = 126$ m, freestream velocity $v_\infty = 11.4$ m/s, angular speed $\Omega = 12.1$ rpm, and air density $\rho = 1.225$ kg/m^3).

Since all turbines use the same section-wise design across all their N_B blades, $T = B$, allowing the design and operation tensors to be indexed simultaneously. This tensor-based framework enables the differentiable optimization pipeline to process multiple blade designs in parallel via GPU acceleration.

3.2 Differentiable Simulation

Our fully differentiable pipeline starts with NeuralFoil [14], a neural network that acts as a differentiable manifold operator and is a surrogate model of XFoil [5] (the state-of-the-art in airfoil aerodynamic simulation) and maps the Kulfan [9] design parameters of the airfoil to its lift (C_L) and drag (C_D) coefficients. NeuralFoil was trained on different operation conditions (angle of attacks and Reynolds numbers), making it a robust estimator of the airfoil aerodynamic coefficients. The results in Figure 1 demonstrate that *NeuralFoil* achieves comparable accuracy while significantly reducing computational cost, making it suitable for gradient-based optimization within the proposed framework.

The aerodynamic coefficients are then passed to the implicitly differentiable BEMT solver based on a single residual formulation developed by [12]. This formulation solves for the inflow angle ϕ directly using the Brent method with guaranteed convergence [4], computing induction factors explicitly without nested

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_4

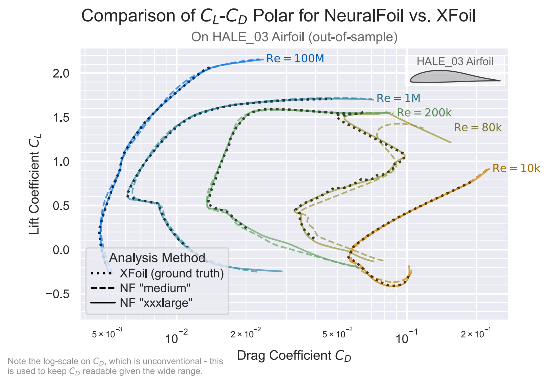


Fig. 1: Comparison between NeuralFoil (NF) and XFoil predictions of aerodynamic coefficients for the HALE03 airfoil at different Reynolds numbers. Source: [14].

iteration, leading to a faster computation compared to traditional BEMT implementations. The BEMT model computes the thrust (T), torque (Q), and power (P) (and its corresponding coefficients C_T, C_Q, C_P) generated by a turbine given its design ($\mathbf{D}$) and operation ($\mathbf{O}$) tensors.

For each radial section, the solver finds the induction inflow angle ϕ that satisfies the residual equation $R(\phi) = 0$ described in Equation 1:

$$R(\phi) = \frac{\sin\phi}{1-a} - \frac{v_\infty}{\Omega r}\frac{\cos\phi}{1+a'} = 0 \tag{1}$$

where $a = \frac{\kappa}{1+\kappa}$ and $a' = \frac{\kappa'}{1-\kappa'}$ are the axial and tangential induction factors, $\kappa = \frac{\sigma C_n}{4F\sin^2\phi}$ and $\kappa' = \frac{\sigma C_t}{4F\sin\phi\cos\phi}$ are the axial and tangential loading parameters computed *explicitly* from ϕ and C_n and C_t are the axial and tangential force coefficients, computed using the lift (C_L) and drag (C_D) coefficients estimated by NeuralFoil. F is the Prandtl tip loss factor and σ is the local solidity.

Crucially, instead of unrolling the iterative solver operations (which is memory-intensive), we compute the gradients of the equilibrium state using the Implicit Function Theorem (IFT), as done in [1] and can be formulated as:

$$\frac{\partial\phi^*}{\partial\boldsymbol{\xi}} = -\left(\left.\frac{\partial R}{\partial\phi}\right|_{\phi=\phi^*}\right)^{-1}\frac{\partial R}{\partial\boldsymbol{\xi}} \tag{2}$$

where $\boldsymbol{\xi}$ denotes the parameters of the residual function (design and operation variables) and ϕ^* is the root of the residual function. This approach decouples the memory cost of gradient computation from the convergence path of the solver, ensuring numerical stability and scalability even when simulating complex blade geometries in parallel. This architecture introduces two paradigm shifts that distinguish it from other solutions for wind turbine design optimization in the industry:

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_4

– **A Table-Free, Dynamic Flow Model:** Unlike traditional BEMT solvers that rely on discrete lookup tables (polars) pre-computed at fixed Reynolds numbers, our approach dynamically queries the neural surrogate at the exact local Reynolds number $\mathrm{Re}(\phi)$ experienced by each blade section during the solver iteration. This eliminates interpolation errors and the "staircase" discontinuities typical of table-based lookups, ensuring that the aerodynamic forces evolve continuously with both the flow state and the blade geometry.
– **End-to-End Differentiable Shape Optimization:** NeuralFoil serves as a fully differentiable operator $\mathcal{N}(\alpha; \mathbf{D}, Re) \rightarrow (C_L, C_D)$. This allows the computation of exact partial derivatives of the aerodynamic coefficients with respect to the airfoil shape parameters $\mathbf{D}$ via automatic differentiation. Consequently, gradients can flow from the global power coefficient C_P back to the fundamental geometric description of the blade, unlocking the ability to optimize the airfoil profiles themselves, rather than merely scaling chord and twist. Moreover, the implicit differentiation of the residual value drastically reduces computational time and complexity.

To benchmark the proposed BEMT framework, we evaluated turbine performance across a range of inflow wind speeds and compared the results against QBlade [11], as shown in Figure 2. Overall, the two models exhibit close agreement throughout the operating envelope. The primary discrepancy arises at low wind speeds (≤ 9 m/s), where QBlade exhibits numerical instability in the induction-factor solution, while the proposed implementation remains stable.

These results highlight the numerical robustness of the single-residual formulation introduced by [12], as well as the dynamic, implicit computation of both the Reynolds number and the induction factors within the solver. This formulation enables a consistent and smooth evaluation of the airfoil lift and drag coefficients across varying flow regimes, without relying on pre-tabulated polars or iterative sub-solvers.

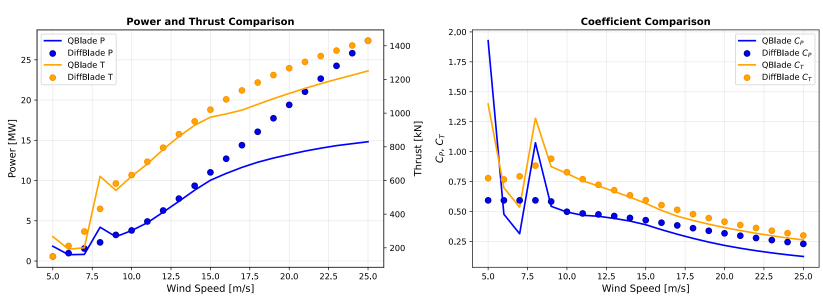


Fig. 2: Comparison of BEMT results from this work (DiffBlade) and QBlade.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_4

3.3 Lagrangian Design Optimization

In order to optimize the blade design ensuring its viability, it is necessary to use constraints regarding both the design and operation of the blades. However, embedding these constraints in the design parameters is not possible, since there is not a direct mapping from Kulfan parameters to design validity. To solve this problem, we propose a blade design process formulated as a constrained nonlinear Lagrangian optimization problem. Our objective is to maximize the rotor power coefficient C_P with respect to the design tensor $\mathbf{D}$, subject to both physical and geometric constraints. This optimization problem is an Augmented Lagrangian Optimization (ALO) [2]. Unlike simple penalty methods, which require the penalty coefficients to approach infinity (leading to Hessian ill-conditioning and numerical instability), ALO converges to the exact constrained solution with finite penalty weights by introducing explicit Lagrange multipliers [13]. The ALO problem can be formally described as:

$$\min_{\mathbf{D}} \mathcal{L}(\mathbf{D}) = -C_P(\mathbf{D}) + \lambda_s h_s + \frac{\mu_s}{2} h_s^2 \tag{3}$$

where h_s is a differentiable geometric feasibility measure that quantifies how far a blade airfoil shape deviates from basic geometric validity requirements (leading and trailing edge thickness and no surface overlap), with associated Lagrange multiplier λ_s and penalty weight μ_s.

The optimization proceeds in a dual-loop architecture: The **inner (primal) loop** minimizes $\mathcal{L}$ with respect to $\mathbf{D}$ using gradient descent optimizers (e.g., Adam or L-BFGS). The gradients $\nabla_{\mathbf{D}}\mathcal{L}$ are computed exactly via the differentiable pipeline, combining the IFT-based aerodynamic gradients $\nabla_{\mathbf{D}} C_P$ with the analytical gradients of the geometric constraints $\nabla_{\mathbf{D}} g$. The **outer (dual) loop** updates the multipliers λ using the rule $\lambda_i \leftarrow \lambda_i + \mu_i \max(0, g_i(D))$ and adaptively increases μ if constraint violations do not decrease sufficiently.

This dual-loop strategy enables the simultaneous optimization of aerodynamic performance and geometric feasibility within a unified, fully differentiable framework. By decoupling the primal design updates from the dual constraint enforcement, the proposed ALO formulation avoids the numerical ill-conditioning typically associated with pure penalty methods while ensuring convergence to constraint-satisfying solutions. Moreover, the use of smooth, physically motivated constraint functions allows the optimizer to exploit accurate gradient information throughout the design space, facilitating efficient exploration of high-dimensional blade geometries.

Figure 3 presents the results of the Augmented Lagrangian Optimization applied to the NREL 5MW reference turbine. The proposed framework achieves an improvement in the rotor power coefficient C_P, from 50.05% to 51.39%. This performance gain is obtained while maintaining low geometric constraint violations and moderate values of the Lagrange multipliers, indicating effective constraint enforcement without excessive penalization.

The hub, mid-span, and tip airfoil sections of the optimized blade are shown in Figure 4. The resulting geometries satisfy the imposed geometric feasibility

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_4

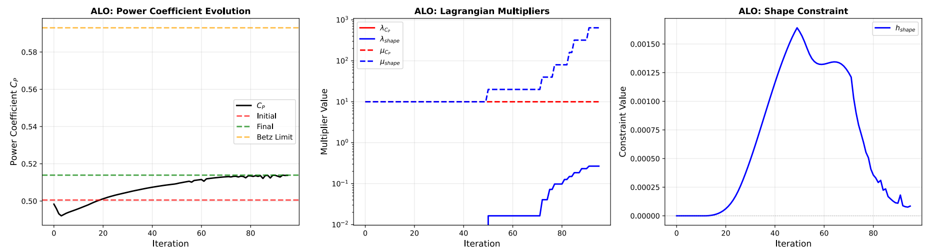
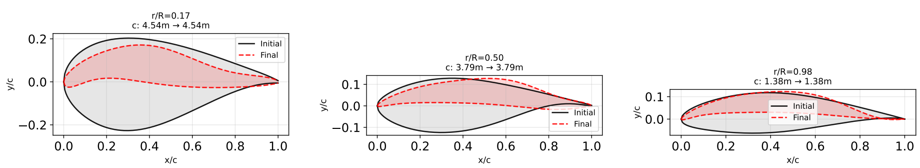


Fig. 3: C_P evolution (left), Lagrange multipliers (center), and shape penalty (right) during the ALO of the NREL 5MW turbine.

criteria: the leading and trailing edges are preserved, and no overlap occurs between the upper and lower surfaces, while reducing airfoils surface area (and consequently manufacturing costs) and improving its efficiency. These results confirm that the optimization process successfully improves aerodynamic performance while maintaining valid airfoil shapes throughout the blade span. It should be noted that the adopted geometric validation does not explicitly account for manufacturing constraints, which could be incorporated.



(a) Hub ($r/R = 0.17$) (b) Mid ($r/R = 0.50$) (c) Tip ($r/R = 0.98$)

Fig. 4: Initial (standard) and final optimized designs of representative airfoil sections along the NREL 5MW turbine blade span.

4 Conclusions

When applied to the NREL 5MW reference turbine, the proposed solver demonstrated improved numerical robustness compared to QBlade, particularly at low wind speeds. The optimization increased the rotor power coefficient C_P from 50.05% to 51.39% while maintaining low shape constraint violations and moderate Lagrange multipliers.

By enabling direct optimization of airfoil geometries, rather than only chord and twist distributions, the framework allows richer aerodynamic design exploration. In summary, this implicitly differentiable, physics-informed framework provides a scalable and robust pathway to optimize wind turbine blades end-to-

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_4

end, bridging the gap between surrogate modeling and physics-based simulation while enabling practical constrained design improvements.

Future work will incorporate structural constraints and further validate the approach against high-fidelity CFD simulations.

Acknowledgments. The authors gratefully acknowledges the financial support provided by the São Paulo Research Foundation (FAPESP), grant number 2025/18319-0. This study was also supported by the PRH 04 program and the National Agency of Petroleum, Natural Gas and Biofuels (ANP).

References

1. Amos, B., Kolter, J.Z.: Optnet: Differentiable optimization as a layer in neural networks. In: International Conference on Machine Learning. pp. 136–145. PMLR (2017)
2. Bertsekas, D.P.: Nonlinear Programming. Athena Scientific, Belmont, MA (1997)
3. Bottasso, C.L., Campagnolo, F., Croce, A.: Optimization of wind turbines using scaled model experiments and numerical simulations. Renewable Energy **40** (2012)
4. Brent, R.P.: Algorithms for minimization without derivatives. Prentice-Hall, Englewood Cliffs, NJ (1973)
5. Drela, M.: Xfoil: An analysis and design system for low reynolds number airfoils. In: Low Reynolds number aerodynamics. pp. 1–12. Springer (1989)
6. Giles, M.B., Pierce, N.A.: An introduction to the adjoint approach to design. Flow, turbulence and combustion **65**(3), 393–415 (2000)
7. Jameson, A.: Aerodynamic design via control theory. Journal of scientific computing **3**(3), 233–260 (1988)
8. Kochkov, D., Smith, J.A., Alieva, A., Wang, Q., Brenner, M.P., Hoyer, S.: Machine learning–accelerated computational fluid dynamics. Proceedings of the National Academy of Sciences **118**(21) (2021)
9. Kulfan, B.M.: Universal parametric geometry representation method. Journal of Aircraft **45**(1), 142–158 (2008)
10. Li, J., Du, X., Martins, J.R.: Deep learning for rapid prediction of the aerodynamic performance of airfoils. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering (2018)
11. Marten, D., Wendler, J., Pechlivanoglou, G., Nayeri, C., Paschereit, C.: Qblade: an open source tool for design and simulation of horizontal and vertical axis wind turbines. In: International journal of emerging technology and advanced engineering. vol. 3, pp. 264–269 (2013)
12. Ning, A.: A simple solution method for the blade element momentum equations with guaranteed convergence. Wind Energy **17**(9), 1327–1345 (2014)
13. Nocedal, J., Wright, S.J.: Numerical Optimization. Springer Science & Business Media (2006)
14. Sharpe, P.: Neuralfoil: A differentiable airfoil aerodynamics surrogate. arXiv preprint arXiv:25xx.xxxxx (2025), available at https://github.com/peterdsharpe/NeuralFoil

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29912-3_4