

A multi-dimensional machine learning framework for superior propeller design choices

Shubham Shubham* and Ankit Sharma[†]
Cranfield University, Bedford, MK43 0AL, United Kingdom

In this study, PropAI is presented, which is a scalable, multi-dimensional surrogate modelling framework for propeller performance prediction that compiles a large 5D baseline database and couples it to a KD-tree Gaussian radial-basis-function (RBF) interpolator. A full-factorial sweep over five design parameters (rotational speed, freestream velocity, blade pitch, diameter, and number of blades) yields 88,540 operating points evaluated via low-fidelity BEMT in QBlade. The trained surrogate reproduces these data with near-machine precision (global RMSE $\approx 6.2\text{e-}4$, MAE $\approx 1.0\text{e-}5$, $R^2 = 1.000$), and parity plots of predicted vs true thrust and power lie essentially on the 45° line (1:1 line). Cross-validation of leave-one-out (LOO) errors confirms excellent generalisation, with both thrust and power below 0.6% of full-scale output. Integrated with an UNSGA-III optimiser, the model performs multi-objective optimisation of thrust, power, and thrust-to-power ratio, producing Pareto fronts that reveal the trade-offs among these metrics. This lightweight, gradient-capable surrogate enables rapid design-space exploration (e.g. via 1D/2D response slices, pair-plots, parallel coordinates) and provides insight into parametric interactions.

I. Introduction

Propellers sit at the core of emerging electric vertical-take-off-and-landing (eVTOL) aircraft and other advanced-air-mobility (AAM) concepts, where multiple low-diameter propellers must deliver high thrust, low noise and stringent efficiency under low-Reynolds-number conditions. Achieving an acceptable compromise among these competing requirements is intrinsically a multi-objective optimisation problem that cannot be solved by classical one-factor-at-a-time design loops. Recent advances in machine-learning surrogates and evolutionary search algorithms now allow thousands of design variants to be screened in seconds rather than days, enabling rapid exploration of large geometric and operating spaces while capturing the complex aero-acoustic-structural couplings that govern public acceptance. Against this backdrop, a fast, extensible and data-driven framework for propeller optimisation is no longer optional - it is essential for keeping pace with the escalating demand for quiet, efficient, small-to-medium-scale propellers powering the next generation of urban air vehicles.

Previous authors have conducted various studies to understand the effect of different design parameters on propeller design and blade shape optimisation, either using a standalone physics-based model or a data-driven methodology. Chang et al. [1] optimised the twist distribution of propeller blades using non-linear programming. This was done for both General Aviation (GA) and Pursue Model (PM) propellers with efficiency improvement of 6% and 1%, respectively. Cho et al. [2] employed the Extended linear Interior Penalty function Method (EIPM) to optimise the propeller blade shape for efficiency improvement. The method improves the efficiency of propeller blade shapes, especially when using lifting line and lifting surface methods for aerodynamic calculations and shows agreement with other numerical schemes and proving useful for validating propellers and designing high-efficiency propellers. Toman et al. [3] use two aerodynamic models, the Blade Element Momentum (BEM) theory and Computational Fluid Dynamics (CFD), to optimise propeller blade shape. The Sequential Meta-Modelling (SM) concept and Design of Experiments (DOE) algorithm with kriging response surface were employed. Although both techniques yielded similar optimum points, the SM algorithm demonstrated better computational efficiency.

Sullivan [4] studied the effect of sweep on propeller performance. Firstly, the power coefficient (C_P) decreases with sweep at a given advance ratio. Secondly, the power loading decreases in the sweep region. Thirdly, the ideal efficiency remains consistent for equal values of C_P . Bergmann et al. [5] introduce sweep correction models for propeller analysis, including lift, drag, and tangential force corrections. The results show that turning airfoils are essential for swept propeller analysis, and up to 25% sweep, the Blade Element Theory (BET) simulations align reasonably well with

*Post-Doctoral Research Fellow, School of Aerospace, Transport and Manufacturing, shubham.shubham@cranfield.ac.uk

[†]Research Fellow, School of Aerospace, Transport and Manufacturing, ankitsharma@cranfield.ac.uk

RANS simulations. The Busemann lift coefficient correction significantly improves power and thrust prediction. Some authors have also worked on a multi-fidelity approach for propellers to understand their performance ([6–8]), and other authors have worked on aeroacoustic analyses using a single fidelity [9–11].

Furthermore, a number of authors have combined surrogate models to better understand propeller design. Vesting et al. [12] introduced a surrogate-assisted multi-objective optimisation for marine propeller design under cavitation constraints. They compared Kriging and neural-network response surfaces to replace costly CFD evaluations, enabling automated blade optimisation with improved efficiency and reduced cavitation risk. Shubham et al. [13] characterised uncertainty in propeller aerodynamic and aeroacoustic performance using a multi-fidelity Kriging surrogate model and Monte Carlo method. The study showed that uncertainty varies with different values of design parameters (such as different blade sweep), and it is important to optimise uncertainty in addition to optimising physical performance. Geng et al. [14] applied deep reinforcement learning with transfer-learning surrogates to simultaneously improve propeller aerodynamic and acoustic performance. Their multi-fidelity DNN surrogate (integrated in a DDPG framework) cut computational cost by $\sim 77\%$ and found blade designs that reduce noise by ~ 1.7 dB with negligible loss in propulsive efficiency. For more studies on propellers/rotors with surrogate models, the readers are referred to the following articles: Wu et al. [15], Yuepeng et al. [16], and Zuhail et al. [17].

Existing propeller-optimisation studies rely on narrowly scoped surrogate models tied to a single geometry, operating range, and objective set. As a result, each new design effort must repeat the labour-intensive task of generating and validating its own performance database before optimisation can begin, wasting time and knowledge gained. To remove this bottleneck, a framework is developed (named as PropAI) that compiles a large, multi-dimensional baseline database and couples it to a lightweight surrogate that can be trained even on a laptop. The database is intended to serve as a common starting point for any future propeller design or optimisation study, irrespective of the surrogate technique adopted. The framework is extensible to even higher dimensions in the future while still keeping the computational cost sufficiently low. Additionally, using the trained surrogate model, a design space study is conducted to understand propeller performance and the mutual interaction of different design parameters.

The present work constructs a five-dimensional (5D) design space defined by rotational speed (RPM), freestream velocity V_∞ , number of blades, blade pitch, and propeller diameter; chord, twist and airfoil distributions are held fixed. An equidistant full-factorial sweep over these variables yields 88,540 operating points, each evaluated with the blade-element–momentum theory (BEMT) implemented in open-source software QBlade 2.0 [18]. A KD-tree with local Gaussian radial-basis function (RBF) interpolation is then fitted to this data, producing an accurate, gradient-capable surrogate whose training cost scales only with the number of neighbours (K), not with the full data set. Finally, the surrogate is coupled to a UNSGA-III algorithm to demonstrate multi-objective optimisation studies in thrust, power, and thrust-to-power ratio across the entire 5-D space.

The paper is structured as follows. Section II discusses the PropAI framework including propeller geometry, 5D design space, methodologies used in the framework including physics-based models, surrogate modelling and multi-objective optimisation methods. Section III reports all the results obtained in this study, including 1D and 2D design space slices, parity plots, RMS errors in the trained surrogate and Pareto fronts obtained. Finally, Section IV discusses the conclusions of this study and future recommendations.

II. PropAI framework

A. Propeller geometry

The baseline propeller geometry used is a 2-bladed rotor and employs the NACA 4412 airfoil profile throughout the blade span starting from 0.2 to 1 r/R . This is consistent with the propeller geometry used in Casalino et al. [19]. Although Casalino’s geometry has an additional tangential component to different radial positions of the leading and trailing edge, the geometry used in the current study has zero tangential component. The blade geometry is made up of 120 blade sections with varying twist and chord lengths as shown in Fig. 1. The radial variations in chord length and twist of the airfoil sections are shown in Figure 2. The propeller has a diameter D of 0.3 m, tip radius $R_T = 0.15$ m, and a root section that transitions from a NACA 0012 airfoil at a radius of 1 cm ($r/R_T = 6\%$) to NACA 4412 at 0.2 r/R . The hub radius is 1.25 cm.

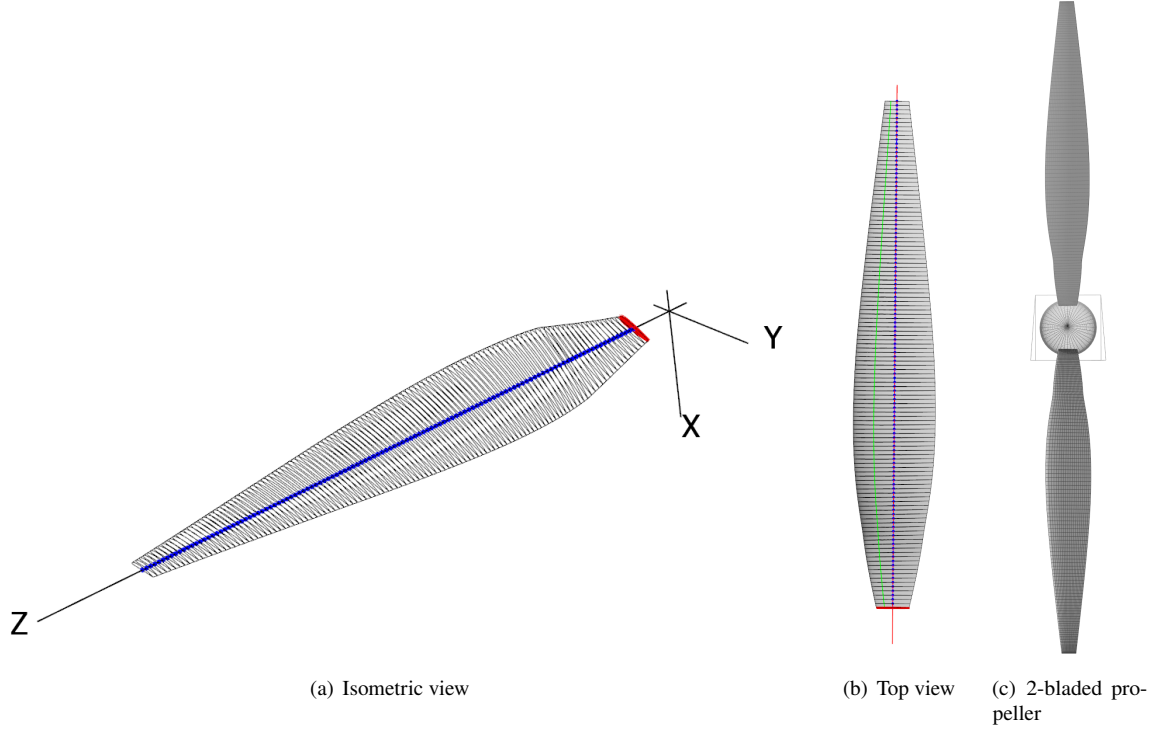


Fig. 1 Baseline propeller geometry

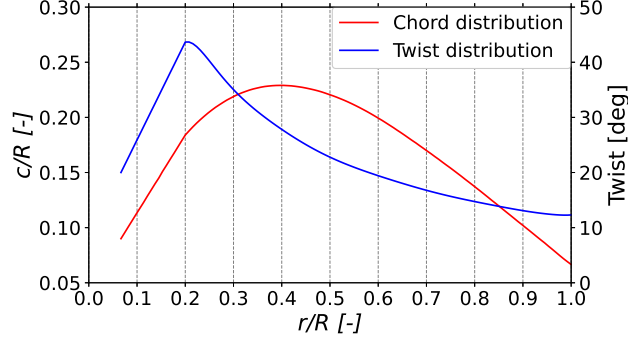


Fig. 2 Chord and twist distribution of the propeller blade

B. Problem definition and design space

The surrogate modelling workflow begins with a carefully defined design space for propeller geometry and operating conditions. The initial design space comprises five key parameters with full 5D factorial sampling:

- Rotational speed (RPM): 0 – 5000, with a step of 500
- Freestream velocity (V_∞): 0 – 30 m/s, with a step of 1 m/s
- Number of blades: 2 – 3 with a step of 1
- Blade pitch angle: 0° – 12° , with a step of 1°
- Propeller diameter: 0.05 – 0.5 m, with a step of 0.05m

These 5D design parameters capture the primary design degrees of freedom at the outset. The continuous (RPM, V_∞ , pitch, diameter) and discrete (number of blades) variables span broad ranges to encompass both low and high operating conditions, while exploring different propeller design configurations. In practice, each sample in this space corresponds to a unique propeller design (geometry and operating point) for which performance can be evaluated using

any CFD method.

In sampling the design space, a physical constraint is imposed, and designs that yield a blade tip Mach number (M_t) greater than 0.4 are discarded a priori. To keep the design space simple in this study, $M_t \leq 0.4$ corresponds to the highest RPM and highest diameter. This ensures that all retained designs operate in a fully subsonic flow regime, well below the onset of any compressibility or local shock effects. For context, traditional propeller design guidelines often keep M_t below roughly 0.75 to avoid shock wave formation on the blades [20]. This constraint also ensures that the resulting surrogate model will not extrapolate into regimes with fundamentally different physics (e.g., high-subsonic or transonic flow). Any future expansion of the design space would be accompanied by a corresponding adjustment of this Mach limit, or a transition to a compressible-flow capable analysis tool.

C. Physics-based models

The current framework is capable of taking multi-fidelity data as input, which can include a combination of either low- and mid-fidelity or mid- and high-fidelity. Although for the current paper, only single-fidelity data is used, employing the low-fidelity Blade Element Momentum Theory (BEMT) using the open-source software QBlade [18].

Flow solver: low-fidelity

The propeller loads were predicted with the Blade Element Momentum Theory (BEMT), originally formulated by Glauert [21], which is a coupling of Actuator Disc theory (Momentum theory) and Blade Element Theory (BET). Momentum theory treats the rotor as an infinitesimally thin actuator disc embedded in a steady, incompressible, axisymmetric, inviscid flow. By applying continuity and the axial momentum balances within a stream-tube, the thrust and power are expressed through the axial induction factor a that quantifies the local velocity deficit. The velocity in the rotor plane as a function of incoming velocity u_∞ is expressed as: $u=(1-a)u_\infty$. The model assumes equal static pressure far upstream and downstream and neglects viscous losses; under these premises, it reproduces the rotor performance coefficients as a function of induction factor that have remained the fundamental reference since the work of Glauert [21] and its modern reformulation by Branlard [22].

Blade element theory resolves the spanwise distribution of aerodynamic load by discretising each blade into independent 2D sections. At every radial station, the local relative flow is constructed from the incoming axial velocity and the rotational component of the wake, yielding an inflow angle that, together with the geometric pitch, fixes the section angle of attack. Airfoil lift and drag coefficients drawn from steady 2D data tables are then used to compute sectional forces. The approach presumes quasi-steady behaviour and ignores spanwise flow, 3D stall delay and tip losses (Leishman [23]).

The classical BEMT couples these two frameworks by enforcing equality between the blade-element loads and the momentum-theory loads within each annular control volume. For every radial element, an iterative fixed-point procedure is executed: initial induction factors are guessed, sectional forces are obtained from the blade equations, revised induction factors (axial a and tangential a') are inferred from the momentum balances, and convergence is declared when successive changes in induction factors fall below a certain value (e.g. 10^{-4}). Finite-blade and rotational effects that lie outside the scope of the original theory are included at each iteration through the Prandtl tip and hub loss factor (Prandtl [24], Glauert [21]) and a 3D stall-delay correction following Snel, Houwink and Piers [25].

To ensure accurate results with both low and mid-fidelity methods, precise airfoil data is essential. To obtain lift and drag coefficients, XFOil [26] is employed for Reynolds numbers ranging from 1×10^5 to 5×10^6 . The XFOil calculations utilize an $NCrit$ value of nine and natural transition at both the pressure and suction side. The airfoil's static polar data is extrapolated to a 360° range of angle of attack (AoA) using the Montgomerie method [27] to ensure a smooth extrapolation in the post-stall regime.

D. Surrogate modelling: KD tree + local RBF

A non-parametric, locally adaptive surrogate model that combines a KD-tree based neighbour search [28] with local radial basis function (RBF) interpolation [29]. A KD tree is a space-partitioning data structure used to organise points in a K -dimensional space. It's essentially a binary tree where each node represents a region in space, and the tree structure is used to efficiently search for points based on their coordinates. The KD tree divides the K -dimensional space into smaller regions, effectively creating a hierarchical structure of these regions. The tree structure is a binary tree, meaning each node has at most two child nodes. KD trees are commonly used for nearest neighbour searches and range searches. Given a query point, the KD tree can efficiently find the closest data point(s) to the query point, and given a region (e.g.,

a rectangle in 2D), the KD tree can efficiently find all data points within that region. The methodology applied in this study is explained below.

1. Data Normalisation and KD-Tree Indexing

All input features are first linearly scaled to the [0,1] range using min–max normalisation. After scaling, we construct a KD-tree on the training sample inputs (with a leaf size of 40 for efficient queries). The KD-tree provides an index for rapid k-nearest neighbour (kNN) search in the 5-dimensional design space [30]. In effect, this is the only “training” required – no global model parameters are fitted in advance, aside from storing the data in the tree. The KD-tree allows neighbour queries in $\sim O(\log N)$ time, making the method scalable to large datasets (with $N=88,540$ here). Five dimensions are well within the regime where exact KD-tree queries significantly outperform brute-force search [30]. This indexing is crucial to swiftly retrieve local data for interpolation at prediction time.

2. Local RBF Surrogate Model Construction

At prediction time, for a given new design point (query), the surrogate model is built on-the-fly using the local neighbourhood of that point. We query the KD-tree for the $K=120$ nearest training samples to the query in the normalized feature space. These K neighbours (and their known thrust/power values) form the support for a localised RBF interpolator. Radial basis function interpolation models the output as a weighted combination of radially symmetric kernel functions centred at the neighbour points [31]. We choose a Gaussian RBF kernel of the form $\phi(r) = \exp(-[r/\sigma]^2)$ where r is the Euclidean distance between the query and a neighbour in feature space. The kernel scale σ is set adaptively for each query as the median distance among the K neighbours, which serves as a local length scale. This distance-based kernel is dimension-independent and meshless, depending only on relative distances, and it yields an exact interpolation scheme known for high accuracy and simplicity [32].

Using the K neighbour points, we construct the RBF interpolation equations requiring the surrogate to reproduce each neighbour’s output exactly. In matrix form, let $A_{ij} = \phi(\|x_i - x_j\|)$ be the $K \times K$ kernel matrix for the neighbours, $x_1, \dots, x_K$. We solve $Aw = y$ for the weight vector w , where y is the vector of the neighbours’ known output values (thrust or power). To prevent numerical ill-conditioning (e.g. if points are very close or collinear in the inputs), a small regularisation term (jitter) is added to the diagonal of A before inverting it. This Tikhonov-style regularisation ensures the system is solvable and also provides a modest smoothing effect. Solving this linear system yields the RBF weights w_i for each neighbour. The target output at the query is then predicted by the RBF interpolator as a sum of weighted kernel contributions:

$$\hat{f}(\mathbf{x}_{\text{query}}) = \sum_{i=1}^K w_i \phi(\|\mathbf{x}_{\text{query}} - \mathbf{x}_i\|), \quad (1)$$

where the summation is done separately for each output quantity (so independent RBF models are built for thrust and for power). By construction, if the query coincides with a training point, the surrogate reproduces its exact output (no training error), and in general, the local RBF will smoothly interpolate between known samples. Notably, radial basis interpolation has been widely adopted in surrogate modelling because it achieves high fidelity to nonlinear data with a straightforward formulation and only a few hyperparameters to choose [32] (in our case, K and the kernel scale, which is set automatically). Unlike a global polynomial or neural network, this local surrogate makes no global functional assumptions, instead adapting to the local shape of the data in different regions of the design space.

By focusing on a neighbourhood of each query and using an exact Gaussian RBF interpolator, this method captures local nonlinear effects with high fidelity. The approach is non-parametric (the model complexity grows with data, but only local subsets are used at a time) and locally adaptive (each region of the design space is modelled with its own localised function), which helps maintain accuracy across a wide domain without needing a globally complex function. Furthermore, the RBF surrogate has the ability to compute analytic gradients, which is a key advantage for design optimisation, as it facilitates gradient-based search or sensitivity analysis. Importantly, the combination of spatial indexing and local modelling makes the surrogate scalable to large datasets – a single global RBF model over all 88k samples would be impractically large to train or invert, whereas this local approach handles it by many lightweight local interpolations.

3. Data and design space checks

Table 1 lists various metrics to perform sanity checks on the 5D surrogate fit. The KD-tree + local RBF surrogate reproduces the 88,540 training samples with machine-precision accuracy: the global root mean squared error (RMSE) is $6.237\text{e-}04$, the mean absolute error (MAE) is $1.048\text{e-}05$, and $R^2 = 1.0$. Such near-zero residuals are expected because the Gaussian RBF interpolant is an exact interpolator and the KD-tree always includes the query point among its neighbours; any remaining error is purely numerical round-off. These statistics confirm that the surrogate has been assembled correctly and can be trusted as a loss-free replacement for the underlying BEMT evaluations when sampling inside the original design grid.

Table 1 Sanity checks for the KD tree + local RBF surrogate fit

Root mean squared error (RMSE)	Mean absolute error (MAE)	R^2 error
$6.237\text{e-}04$	$1.048\text{e-}05$	1.0

Table 2 lists the global extrema of different performance metrics in the 5D design space. The extrema extracted from the surrogate surface are physically consistent: peak power, thrust, and torque all occur at the most aggressive operating point (RPM = 5000, $V_\infty=0$, pitch = 12° , diameter = 0.50 m, 3 blades). This reflects the expected scaling of aerodynamic loads with disk area, pitch and rotational speed. Zero values for the same metrics appear at the opposing corner (RPM = 0), showing that the database correctly captures the idle condition. Maximum thrust-to-power ratio (5.52 N/W) is attained at a modest RPM of 500 with the smallest diameter (0.05 m) and zero pitch. The result highlights the high specific efficiency of lightly-loaded, slowly-spinning propellers - information that would be lost if only maximum-thrust designs were examined. Finally, minimum T/P appears where pitch is large (10°) and the small-diameter rotor is run at higher RPM, a regime dominated by profile drag, hence poor propulsive efficiency. Some of these extreme values are also shared between different design points; only the first observed points (for each extremum) are listed below.

Table 2 Global extrema and where they occur

Metric	Extreme	Value	RPM	V_∞	Pitch	Diameter	Blades
Power	max	5,384.513	5,000.0	0.0	12.0	0.5	3
Power	min	0.0	0.0	0.0	6.0	0.05	2
Thrust	max	201.589	5,000.0	0.0	12.0	0.5	3
Thrust	min	0.0	0.0	0.0	6.0	0.05	2
Torque	max	10.284	5,000.0	0.0	12.0	0.5	3
Torque	min	0.0	0.0	0.0	6.0	0.05	2
T/P	max	5.516	500.0	0.0	0.0	0.05	2
T/P	min	0.0	1000.0	0.0	10.0	0.05	2

E. Multi-Objective Optimisation

This study uses the NSGA-III (Non-dominated Sorting Genetic Algorithm III), which is a recent extension of the popular NSGA-II method originally developed for handling multi-objective optimisation problems [33]. Unlike its predecessor, NSGA-III is specifically designed to tackle many-objective optimisation scenarios where the number of objectives (e.g., four or more) makes conventional crowding-distance mechanisms less effective [34].

NSGA-II introduced the idea of fast non-dominated sorting, elitism, and crowding distance to maintain a diverse set of solutions. However, as the dimensionality of the objective space grows, relying solely on crowding distance becomes insufficient for preserving population diversity [33]. NSGA-III addresses this challenge by incorporating a set of predefined reference points (or reference lines) in the objective space to guide the selection of offspring [34]. These reference points act as “anchors” that encourage the search to explore under-represented regions of the Pareto front, ultimately improving both convergence speed and solution distribution in higher-dimensional problems.

During each generation, NSGA-III applies non-dominated sorting similar to NSGA-II but leverages the reference points to perform niche-preservation. This mechanism systematically checks how well solutions align or cluster around these reference points, ensuring that the algorithm maintains diverse trade-offs among the objectives [34]. As

a result, NSGA-III can more reliably discover a spread of Pareto-optimal solutions even when the dimensionality of the objectives grows beyond three or four. In practical applications, NSGA-III has proven beneficial over NSGA-II for many-objective problems in engineering, logistics, and data analytics, making it a robust choice for advanced multi-objective optimisation tasks.

III. Results

A. Validation study

Figure 3 benchmarks the BEMT predictions against reference data spanning different fidelity levels: high-fidelity Lattice-Boltzmann Method ("LBM-ref"), mid-fidelity Vortex-Lattice Method ("VLM-ref"), and mid-fidelity Lifting-Line Free-Vortex Wake method ("LLFVW-ref") from Sharma et al. [35], as well as wind-tunnel measurements from Casalino et al. [19]. Across the full advance ratio range $J=0.0-0.8$, BEMT reproduces the global thrust and torque trends with the correct slope and intersection point where thrust (or torque) falls to near zero. The only noticeable bias appears in the very low advance ratio regime ($J \leq 0.2$), where BEMT slightly over-predicts both thrust and torque; this is expected because BEMT neglects viscous losses and three-dimensional tip effects that are most pronounced in near-static conditions. Beyond $J \approx 0.3$, BEMT converges toward all other methods and remains within the spread of experimental and numerical references.

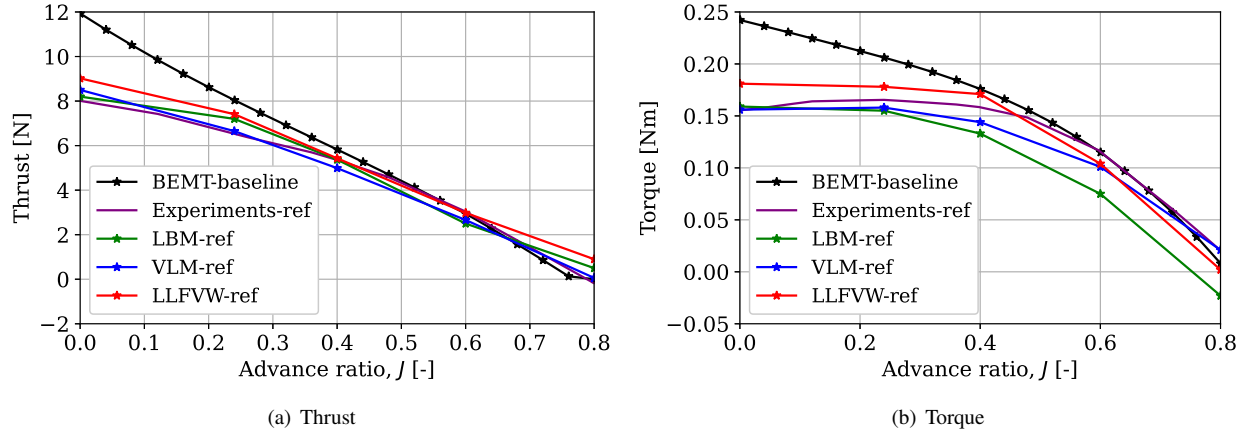


Fig. 3 Validation of BEMT data with experiments [19], high-fidelity LBM, mid-fidelity VLM and mid-fidelity LLFVW methods [35]

These results confirm that BEMT captures the first-order aerodynamic load dependence on advance ratio, providing sufficiently accurate, smoothly varying data for surrogate-model training. Minor low- J discrepancies are acceptable for the present study because the objective is not to refine BEMT itself but to demonstrate how large, five-dimensional simulation datasets can be handled by the proposed KD-tree + local-RBF surrogate framework.

B. Design space exploration and performance metrics

A scatter plot matrix (pair-plot) was generated to visually assess the coverage and correlations of the input design space, as shown in Figure 4. The sampling points are uniformly distributed across each dimension and every two-dimensional projection, typical of a full-factorial sampling. There are no void regions in the plots – every allowable combination of inputs appears as a point in the respective scatter subplots, confirming that the full range of each parameter has been exhaustively combined with the others. For example, when plotting RPM vs. V_∞ , vertical stripes of points are observed at each discrete RPM level (since RPM is taken in increments of 500), and similarly, pitch angle vs. V_∞ shows vertical bands for each 1° pitch increment.

The diagonal panels of the pair plot (often depicting histograms or density for each variable) further illustrate the even design space coverage. In this case, they would show a roughly uniform distribution for each continuous variable (e.g., 11 equally populated bins for RPM from 0 to 5000), and two distinct peaks for the number of blades corresponding

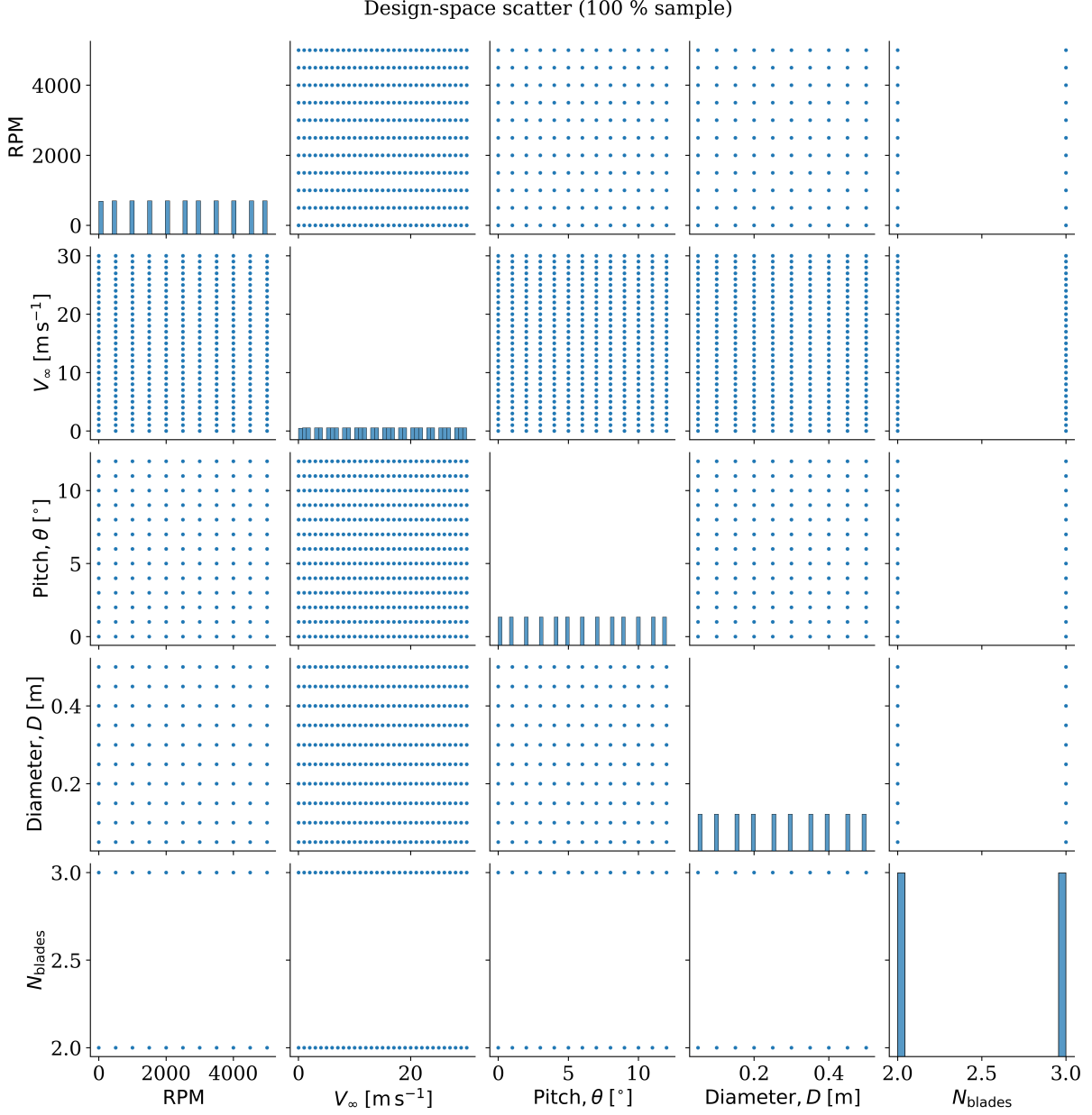


Fig. 4 Scatter matrix (“pair-plot”) of the five input variables in the design space (using the full grid of sampled points).

to the two discrete levels. The symmetry about the diagonal of the scatter matrix (a mirror image layout of plots) provides a convenient check that each pairwise relationship has been examined. Importantly, the pair plot reveals no strong linear correlations between any two input variables, aside from the inherent structured alignment. Because the experimental design varied all factors independently, there is no coupling between inputs (for instance, high RPM values occur across all velocities and diameters, rather than only at certain subsets). This absence of imposed correlation is reflected by the evenly filled point clouds rather than sloped or curved trends. In other words, the design space sampling is orthogonal. This verification step is crucial before proceeding to modelling, ensuring that any patterns found by the surrogate will stem from the physics of the problem rather than gaps or biases in the training data.

To gain further insight into how the design inputs influence the outputs, a parallel coordinates visualisation is employed, as shown in Figure 5. Each high-dimensional data point is represented as a polyline that traverses a series

of parallel vertical axes, one for each variable (here we include all five input parameters, followed by the four output metrics) to visualise multivariate relationships. Colour encoding is applied for the blade count to differentiate the two categories of designs (2 vs 3 blades): orange and blue lines are for the 2-bladed and 3-bladed configurations, respectively. For thrust, torque and power, blue lines extend to higher values than the orange lines, suggesting that for otherwise similar conditions, a 3-bladed rotor generates greater thrust than a 2-bladed rotor, but also consumes more torque and power. In case of thrust/power ratio, 2-bladed designs cluster towards slightly higher values (indicating a marginally higher efficiency for 2 blades compared to 3 blades) in certain speed/rpm/pitch/diameter regimes, which hints that adding a third blade can incur diminishing returns in efficiency under some conditions.

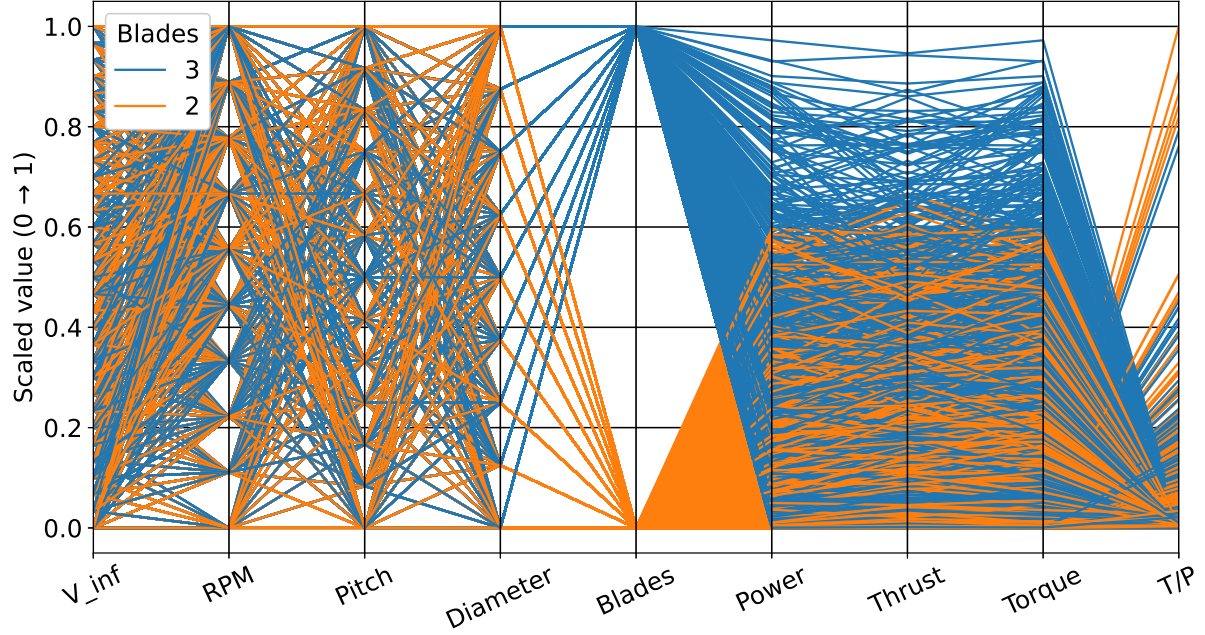


Fig. 5 Parallel coordinate plot of the design space augmented with performance metrics. Each polyline represents one design point, intersecting each vertical axis at the value corresponding to that design’s parameters or results.

In terms of interactions and clusters, multiple lines bundle together for different input design parameters, indicating the factorial grid spacing. The absence of any completely isolated outlier lines in the plot suggests there are no singular anomalous designs – each design’s performance trends smoothly relative to others, without any physically implausible results. Since blade count has a systematic influence on all performance metrics considered, this is a key insight for surrogate modelling; it suggests that the model may need to account for blade count as a significant categorical input, potentially even training separate surrogate sub-models. KD tree + local RBF model fits well within this context. Furthermore, since the 3-bladed propeller produces a wider range of performance values, the surrogate needs to model a more complex relationship (due to more variability and widely fanned-out lines) than for the case of the 2-bladed propeller.

Figure 6 shows the testing of the surrogate model for power and thrust values, against 1D slices of BEMT data for Pitch. Here pitch angle is varied (0° – 12°) while $\text{RPM} = 500$, $V_\infty = 0$ (hover), diameter = 0.30 m, and blades = 2 are held fixed. The surrogate (orange line) forms a smooth, continuous curve and demonstrates an excellent fit to the data (blue dots), reflecting the property of RBF interpolation to smoothly interpolate multi-dimensional data [36] with no visible systematic bias, indicating negligible residual errors. This agreement is expected, as RBF-based surrogates are designed to produce an interpolation surface that is smooth yet non-oscillatory [37], avoiding spurious deviations between data points.

Both power and thrust values rise monotonically with increasing pitch angle. At low pitch, both the BEMT data and surrogate show low power and thrust (on the order of 0.125 W and 0.12 N, respectively). As pitch increases, the power and thrust increase steadily. Notably, the curve begins to level off slightly at the highest pitch angles (around 10 – 12°), when flow separation becomes dominant and blade efficiency decreases. The surrogate reproduces this nonlinearity

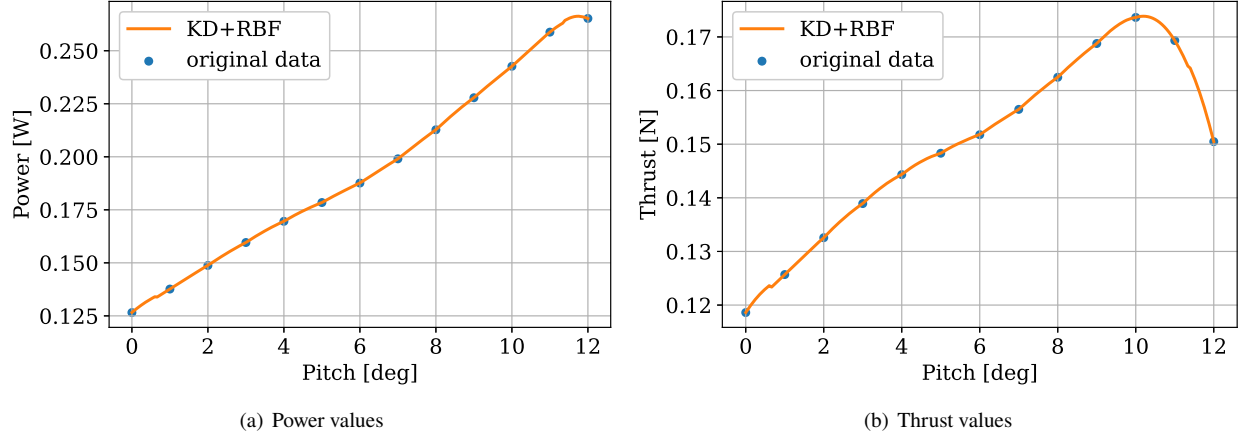


Fig. 6 Power and thrust vs pitch angle at fixed RPM = 500, $V_\infty = 0$ m/s (hover), diameter = 0.30 m, blades = 2. Blue dots are original BEMT results, and the orange curve is KD-Tree + local RBF surrogate prediction.

(especially for thrust), suggesting it generalises the physics well rather than imposing a strictly linear rise. The location and magnitude of the thrust peak (approximately 0.174 N at 10° pitch for this condition) in the surrogate prediction match the simulation data well, indicating the model has captured the underlying nonlinear aerodynamics (stall onset) without smoothing it out. There is no indication of any bias (the surrogate is neither consistently above nor below the blue points), and no significant interpolation error can be discerned – each BEMT data point lies essentially on the predicted curve.

Figure 7 shows how the interpolated response fields (power and thrust) vary over two different 2D slices of the 5-dimensional design space, with white dots indicating the discrete training samples. The first two images have fixed pitch = 0° , diameter = 0.40 m, blades = 2, while the next two images have fixed RPM = 2000, $V_\infty = 0$ (hover), blades = 2.

For the first two images (a and b), the overall trends captured by the surrogate align well with physical expectations: power and thrust both increase strongly with RPM and decrease with forward speed. At the bottom of the contour (near $V_\infty = 0$, i.e. hover), increasing RPM produces a nearly proportional increase in both thrust and power. Along the vertical direction (increasing V_∞), the thrust contours drop off, especially at high RPM; the colour transitions from warm (high thrust) to cooler (lower thrust). The local RBF model seems to cause no spurious oscillations between design points, which can sometimes plague interpolation schemes; the RBF method’s inherent tendency to minimise oscillation yields a nicely behaved, physically coherent surface [36]. The absence of any strange contour shapes or “islands” indicates that the surrogate model is not inventing non-physical behaviour between the training points.

This surrogate behaviour is similar for the second two images (c and d), which shows that increasing the diameter (moving upwards in the plot, from 0.05 m to 0.5 m) dramatically increases both thrust and power, as expected because a larger disk area captures more air mass flow (thrust scales with disk area and flow momentum change). Meanwhile, varying the pitch angle (horizontal axis, 0° to 12°) produces effects on thrust and power, similar to Figure 6. The surrogate plot shows thrust reaching a maximum at an intermediate pitch ($\sim 8\text{--}10^\circ$ across much of the diameter range), beyond which the thrust declines slightly (colours become cooler at the far right edge around 12°). Similar behaviour is found for power values, although the decrease after the peak value is not as significant as the thrust. This is consistent with the physical phenomenon of aerodynamic stall: at high pitch angles, the blade sections exceed their critical angle of attack, reducing lift (and thus thrust). Notably, the peak behaviour around $8\text{--}10^\circ$ pitch shifts insignificantly with diameter.

Figure 8 shows the parity plot of predicted vs true power and thrust values for all 88,540 propeller design points in the 5D design space. The dashed 1:1 diagonal line represents perfect agreement (Predicted = True). All points cluster tightly along this 45° line, indicating a near-unity linear correlation and high fidelity of the KD-tree + local RBF interpolation model across the full output range (approximately 0–5.2 kW for power and 0–220 N for thrust). The predictions show minimal scatter or deviation from the ideal line, with no noticeable systematic bias even at extreme values. This demonstrates that the interpolator can almost exactly reproduce the known data, exhibiting excellent qualitative accuracy and an essentially one-to-one correspondence between predicted and actual performance values.

To estimate the surrogate’s confidence in its prediction, a local uncertainty indicator based on leave-one-out (LOO)

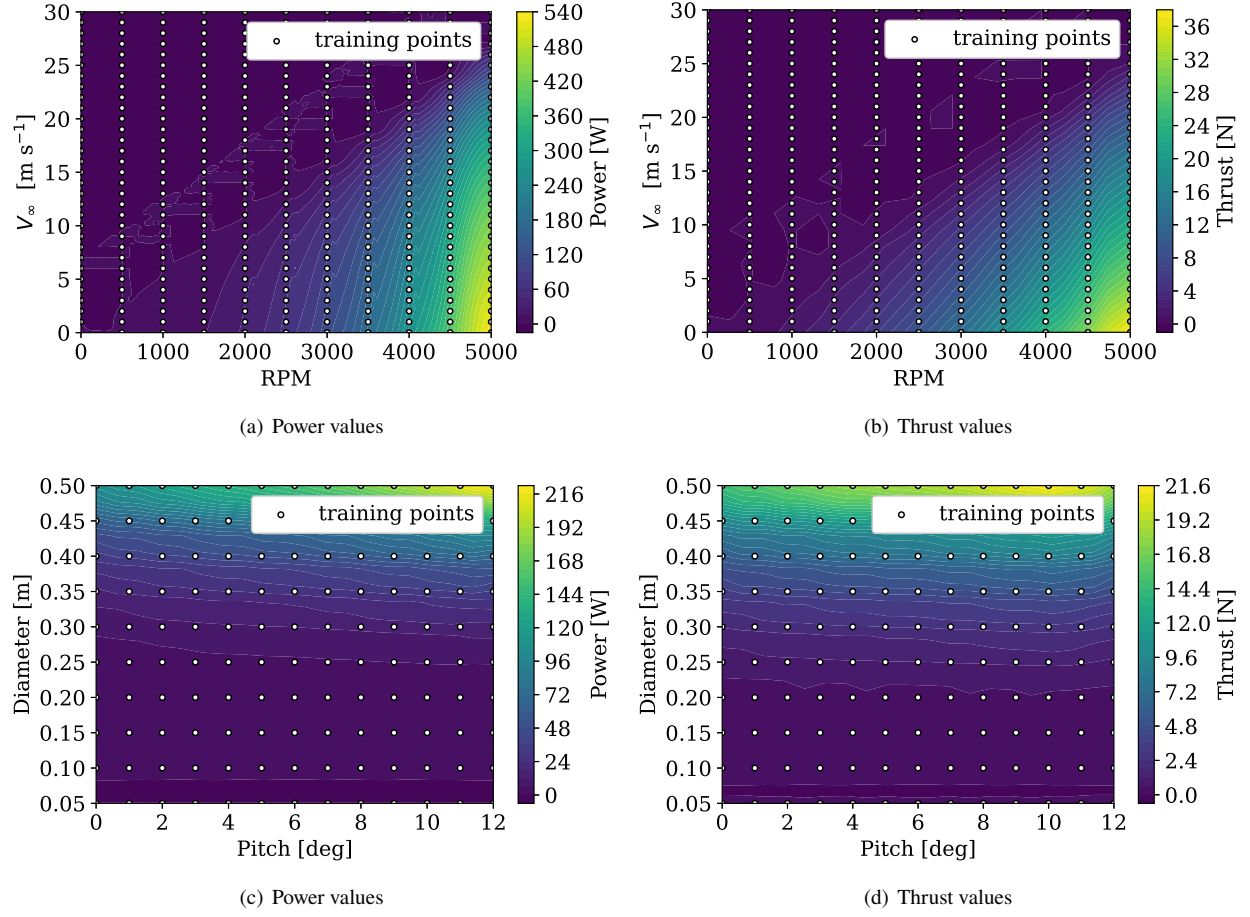


Fig. 7 Contour plots of surrogate-predicted power and thrust over two 2D design space slices, with training data points shown as white circles.

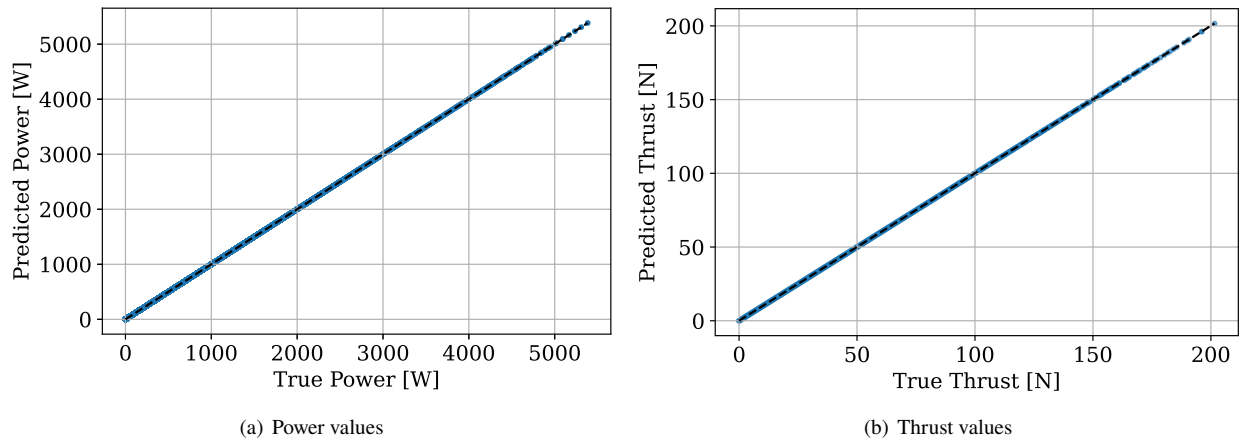


Fig. 8 Parity plot of predicted vs true power (left) and thrust (right) for all 88,540 propeller design points.

cross-validation of the RBF fit is computed. Specifically, after building the local RBF interpolator on the K neighbours, we assess how well it predicts each neighbour when that neighbour's data is withheld from the fit. This provides

K leave-one-out (LOO) residuals (the differences between each omitted neighbour's true value and the surrogate's prediction for it). Root-mean-square (RMS) of these K residuals as a scalar uncertainty estimate is associated with the query prediction. A small LOO error indicates that the local surrogate surface passes close to all the neighbour data points (high confidence), whereas a large LOO residual suggests the query lies in a less certain extrapolation region or that the neighbours' values are difficult to interpolate smoothly (lower confidence).

Figure 9 shows the LOO-RMSE of the model's predictions for power and thrust, shown over the RPM- V_∞ plane (with Pitch = 6° , Diameter = 0.25 m, Blades = 2). The dark purple regions denote very low error, and bright green/yellow regions denote higher error. The interpolation model achieves very low errors (essentially near zero, e.g. only a few Watts or less than 0.2 N) throughout most of the central, well-sampled domain, demonstrating strong generalisation capability where the underlying performance surface varies smoothly. Higher errors are observed toward the boundaries of the design space, especially at the extreme high RPM (~ 5000) and high free-stream velocity V_∞ (~ 30 m/s) corners, indicating weaker generalisation in these regions of steep performance gradients. Even in these edge areas, the maximum LOO-RMSE remains modest (on the order of 32 W for power and 1.3 N for thrust, roughly 0.6% of the respective full-scale outputs), suggesting that the interpolator maintains good overall accuracy while understandably being least confident at the fringes of the sampled space.

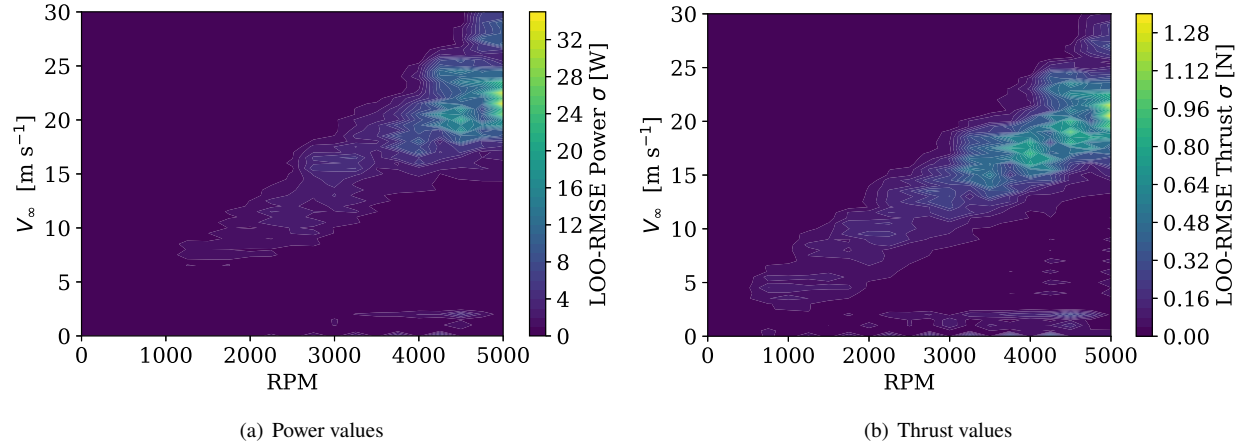


Fig. 9 Leave-One-Out Root Mean Square Error (LOO-RMSE) of the model's predictions for power (left) and thrust (right), shown over the RPM- V_∞ plane (with Pitch = 6° , Diameter = 0.25 m, Blades = 2 fixed).

Figure 10 examines two multi-objective optimisation cases to understand the trade-offs between different parameters: maximise thrust while minimising power, with a velocity constraint $V_\infty \leq 5$ m/s, and maximise the thrust-to-power ratio (an efficiency metric) while minimising power, under a stricter $V_\infty \leq 1$ m/s constraint. The results are obtained using the UNSGA-III algorithm and the trained KD tree + local RBF model on the 5D design and operational parameters. The negative values pertain to the parameter which is maximised. Tables 3 and 4 report the 5D design parameters for the extreme ends of the two Pareto fronts.

Table 3 5-dimensional design parameters for the extreme ends of the Pareto-optimal solutions, for thrust vs power case

RPM	V_∞ [m/s]	Pitch [$^\circ$]	Diameter [m]	Blades	Power [W]	Thrust [N]
634.381	4.871	0.428	0.44	3	2.685	0.321
3113.722	0.112	1.636	0.41	3	273.03	25.089
2979.185	0.233	0.08	0.494	3	485.183	40.935
4964.079	0.107	0.0059	0.492	3	2025.924	103.677
4819.92	1.596	8.708	0.489	3	3512.938	145.371

In Figure 10 (a), the Pareto front shows the expected inverse relationship between thrust and power: as thrust output increases, the required power also increases. The non-dominated solutions start near the origin (low power, low thrust)

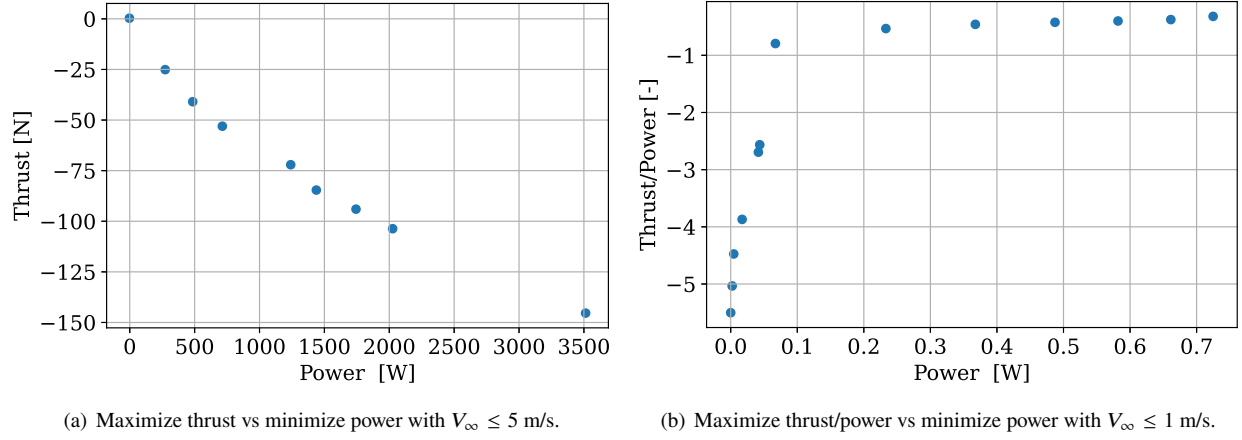


Fig. 10 Pareto-optimal solution sets (Pareto fronts) obtained from the UNSGA-III multi-objective optimisation.

Table 4 5-dimensional design parameters for the extreme ends of the Pareto-optimal solutions, for thrust/power vs power case

RPM	V_∞ [m/s]	Pitch [$^\circ$]	Diameter [m]	Blades	Power [W]	Thrust/Power [-]
500.237	0.0103	0.0799	0.05	2	1.92E-05	5.502
500.021	0.206	0.222	0.05	2	0.00217	5.034
500.028	0.227	0.244	0.05	3	0.0045	4.475
4765.236	0.0101	7.584	0.067	3	0.661	0.375
4761.145	0.615	7.499	0.072	3	0.725	0.319

and extend toward solutions with high thrust (~ 150 N) at high power (~ 3.5 kW), as also seen in Table 3. The shape of this Pareto curve is strongly concave, indicating diminishing returns in thrust for additional power expenditure. For example, a design around ~ 1 kW might produce roughly 70–80 N of thrust, whereas quadrupling the power to ~ 4 kW only improves thrust to ~ 150 N (a roughly 2x thrust gain for 4x power increase). This reflects the physics of propellers: pushing toward maximal thrust becomes inefficient, as much of the extra power is wasted for comparatively small thrust improvements. Furthermore, slightly more points are concentrated near the “knee” region (~ 2 kW) – a point of good compromise where adding more power starts yielding much smaller returns in thrust. This implies that intermediate designs in that gap are either infeasible due to the constraint or strongly dominated.

In Figure 10 (b), the Pareto front looks markedly different from the first case, since high thrust per unit power (propulsively efficient operation) generally occurs at very low power (and thrust) levels; all solutions use < 1 W of power, a tiny fraction of the power in the first case. As power requirement increases, the achievable thrust-to-power ratio declines steeply and trying to get more thrust out of the system forces a sacrifice in efficiency. Quantitatively, comparing the extreme ends of the Pareto front, by allowing half a watt of power consumption, the efficiency drops roughly fivefold compared to the low-power optimum. The Pareto front here is slightly disconnected or multimodal (not continuous), consisting of two distinct clusters of solutions: one cluster is at the extreme lower-left, and another cluster is the extreme upper-right. Such disconnections occur due to the problem’s constraint (V_∞) or physics creating multiple optimal regimes. The first cluster is similar to high-bypass propulsion for hover, where every watt is used to produce as much thrust as possible, but total thrust is inherently limited (in this case, close to 0), so practically infeasible. The second cluster concedes some efficiency to push thrust output closer to the feasible maximum under 1 m/s conditions (by increasing disk loading or RPM, as seen in Table 4, which yields more thrust but also more losses).

IV. Conclusions and Discussions

This work introduces PropAI, a five-dimensional, full-factorial database and a KD-tree + local Gaussian RBF surrogate that together provide a single, reusable baseline for propeller analysis and optimisation. A total of 88,540

BEMT evaluations spanning RPM, freestream velocity, blade count, pitch, and diameter were indexed in a KD-tree; at query time, the 120 nearest neighbours are interpolated with an adaptive RBF kernel. Because the query point is always part of the neighbour set, the surrogate reproduces the training data with machine precision (global RMSE $\approx 6.2\text{e-}4$, MAE $\approx 1.0\text{e-}5$, $R^2 = 1.000$). Leave-one-out cross-validation confirms excellent generalisation: over the entire RPM- V_∞ plane, the worst-case LOO-RMSE is < 32 W for power and < 1.3 N for thrust—both below 0.6% of full-scale output—while the vast majority of the space remains at sub-watt and sub-0.2 N error levels. 1D slice plots and 2D contours show the surrogate captures non-linear stall roll-off and diameter scaling without spurious oscillations, demonstrating that local interpolation plus an adaptive length-scale is sufficient to capture complex physics.

Coupling the surrogate to UNSGA-III revealed clear trade-offs between thrust, power and efficiency. For $V_\infty \leq 5$ m/s, thrust gain beyond ~ 100 N demands disproportionate power (> 2 kW), while under the stricter $V_\infty \leq 1$ m/s constraint efficiency peaks at ~ 5.5 N/W but absolute thrust saturates near 1 N. These Pareto fronts, obtained in minutes on a laptop, illustrate how the framework can guide concept selection and sizing decisions at virtually zero additional simulation cost. Analytic gradients delivered by the RBF kernel further enable sensitivity studies or gradient-based refinement without re-training.

The study demonstrates that a lightweight, locally adaptive surrogate can replace tens of thousands of BEMT calls with negligible loss of fidelity, offering a practical pathway to rapid, multi-objective propeller design. No special hardware is required; training is limited to a single KD-tree build, and the cost of neighbour queries scales as $O(\log N)$, making the approach immediately portable to larger datasets.

Future work will extend the design space to include chord, twist, and sweep distributions as independent variables, raising the dimensionality to 10–15, and will include mid- and high-fidelity CFD or aeroacoustic data via the same KD-tree-RBF machinery to create hierarchical, fidelity-adaptive surrogates. Active-learning loops that target regions of high LOO-error and multi-objective expansions to noise, vibration, and structural metrics are also envisioned. This framework will also be integrated into a web-based visual analytics framework [38] to better analyse the input data and results obtained. Collectively, these steps will evolve PropAI from a five-parameter proof-of-concept into a comprehensive, multi-fidelity design engine for the next generation of low-Reynolds-number propellers and eVTOL rotors.

Acknowledgments

Special thanks to the Early Career Researchers network (ECRn) at Cranfield University, UK, for providing the conference fund to attend the AIAA Aviation Forum 2025.

References

- [1] CHANG, L., and SULLIVAN, J., “Optimization of propeller blade shape by an analytical method,” *18th Joint Propulsion Conference*, 1982, p. 1125.
- [2] Cho, J., and Lee, S.-C., “Propeller blade shape optimization for efficiency improvement,” *Computers & fluids*, Vol. 27, No. 3, 1998, pp. 407–419.
- [3] Toman, U. T., Hassan, A.-K. S., Owis, F. M., and Mohamed, A. S., “Blade shape optimization of an aircraft propeller using space mapping surrogates,” *Advances in Mechanical Engineering*, Vol. 11, No. 7, 2019, p. 1687814019865071.
- [4] Sullivan, J., “The effect of blade sweep on propeller performance,” *10th Fluid and Plasmadynamics Conference*, 1977, p. 716.
- [5] Bergmann, O., Möhren, F., Braun, C., and Janser, F., “Aerodynamic Analysis of Swept Propeller with BET and RANS,” *DLRK 2021*, 2021.
- [6] Li, Z., Maltsev, V., Modarres Aval, A. H., and Da Ronch, A., “Multi-fidelity aerodynamic methods for the analysis of propeller wing interaction,” 2024.
- [7] Di Pasquale, D., Tarun, T., Shubham, S., Prince, S., and Proenca, A., “COMPUTATIONAL ANALYSIS ON THE EFFECTS OF LEADING EDGE TUBERCLES ON THE PERFORMANCE OF ROTATING PROPELLER,” *34th Congress of the International Council of the Aeronautical Sciences*, 2024.
- [8] Shubham, S., Spinelli, A., and Kipouros, T., “Multi-fidelity multi-disciplinary optimisation of propeller design by visual analytics,” *The Eurographics Association*, 2024.

- [9] Hanson, L. P., Baskaran, K., Pullin, S. F., Zhou, B. Y., Zang, B., and Azarpeyvand, M., "Aeroacoustic and aerodynamic characteristics of propeller tip geometries," *28th AIAA/CEAS aeroacoustics 2022 conference*, 2022, p. 3075.
- [10] Grande, E., Shubham, S., Avallone, F., Ragni, D., and Casalino, D., "Computational aeroacoustic study of co-rotating rotors in hover," *Aerospace Science and Technology*, Vol. 153, 2024, p. 109381.
- [11] Shubham, S., Avallone, F., Grande, E., and Casalino, D., "Computational aeroacoustic investigation of co-rotating rotors for urban air mobility," *Delft University of Technology MS Thesis*, 2020.
- [12] Vesting, F., and Bensow, R. E., "On surrogate methods in propeller optimisation," *Ocean engineering*, Vol. 88, 2014, pp. 214–227.
- [13] Shubham, S., Kipouros, T., Dash, S., and Ianakiev, A., "Uncertainty propagation and management of mixed uncertainties for multi-fidelity multi-disciplinary analysis of propeller with different blade sweep," *AIAA SCITECH 2024 Forum*, 2024, p. 1487.
- [14] Geng, X., Liu, P., Hu, T., Qu, Q., Dai, J., Lyu, C., Ge, Y., and Akkermans, R. A., "Multi-fidelity optimization of a quiet propeller based on deep deterministic policy gradient and transfer learning," *Aerospace Science and Technology*, Vol. 137, 2023, p. 108288.
- [15] Wu, X., Zuo, Z., Ma, L., and Zhang, W., "Multi-fidelity neural network-based aerodynamic optimization framework for propeller design in electric aircraft," *Aerospace Science and Technology*, Vol. 146, 2024, p. 108963.
- [16] Yuepeng, B., Wenping, S., Zhonghua, H., Zhang, Y., and Zhang, L., "Aerodynamic/aeroacoustic variable-fidelity optimization of helicopter rotor based on hierarchical Kriging model," *Chinese Journal of Aeronautics*, Vol. 33, No. 2, 2020, pp. 476–492.
- [17] Zuhail, L. R., Faza, G. A., Palar, P. S., and Shimoyama, K., "Multi-objective kriging-based optimization for high-fidelity wind turbine design," *AIAA Scitech 2019 Forum*, 2019, p. 0539.
- [18] Marten, D., "QBlade: a modern tool for the aeroelastic simulation of wind turbines," 2020.
- [19] Casalino, D., Grande, E., Romani, G., Ragni, D., and Avallone, F., "Definition of a benchmark for low Reynolds number propeller aeroacoustics," *Aerospace Science and Technology*, Vol. 113, 2021, p. 106707.
- [20] Colozza, A., "High altitude propeller design and analysis overview," *NASA/CR*, 1998, pp. 98–208520.
- [21] Glauert, H., "Airplane propellers," 1935.
- [22] Branlard, E., "Wind turbine aerodynamics and vorticity-based methods: Fundamentals and recent applications," 2017.
- [23] Leishman, G. J., *Principles of helicopter aerodynamics with CD extra*, Cambridge university press, 2006.
- [24] Prandtl, L., and Betz, A., *Vier Abhandlungen zur Hydrodynamik und Aerodynamik: (Flüssigkeit mit kleiner Reibung; Tragflügeltheorie, I. und II. Mitteilung; Schraubenpropeller mit geringstem Energieverlust)*, Selbstverlag des Kaiser Wilhelm Instituts für Strömungsforschung, 1927.
- [25] Snel, H., Houwink, R., and Piers, W., "Sectional prediction of 3 D effects for separated flow on rotating blades," 1992.
- [26] Drela, M., "XFOIL: An analysis and design system for low Reynolds number airfoils," *Low Reynolds number aerodynamics*, Springer, 1989, pp. 1–12.
- [27] Montgomerie, B., "Methods for root effects, tip effects and extending the angle of attack range to $\pm 180^\circ$, with application to aerodynamics for blades on wind turbines and propellers," *FOI, Swedish Defence Research Agency, Stockholm, Sweden, Report No. FOI*, 2004.
- [28] Friedman, J. H., Bentley, J. L., and Finkel, R. A., "An algorithm for finding best matches in logarithmic expected time," *ACM Transactions on Mathematical Software (TOMS)*, Vol. 3, No. 3, 1977, pp. 209–226.
- [29] Lowe, D., and Broomhead, D., "Multivariable functional interpolation and adaptive networks," *Complex systems*, Vol. 2, No. 3, 1988, pp. 321–355.
- [30] "KDTree in SciPy," <https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.KDTree.html>, 2025. Accessed: 23 Jun 2025.
- [31] "Radial basis functions," https://smt.readthedocs.io/en/latest/_src_docs/surrogate_models/rbf.html, 2025. Accessed: 23 Jun 2025.

- [32] Liang, J., Li, Z., and Khailah, E. Y., “Research on the modified surrogate model based on local RBF for concrete dam static and dynamic response analysis,” *Frontiers in Earth Science*, Vol. 11, 2023, p. 1125691.
- [33] Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T., “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE transactions on evolutionary computation*, Vol. 6, No. 2, 2002, pp. 182–197.
- [34] Deb, K., and Jain, H., “An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints,” *IEEE transactions on evolutionary computation*, Vol. 18, No. 4, 2013, pp. 577–601.
- [35] Sharma, A., Shubham, S., Riaz, A., and Ianakiev, A., “Multi-fidelity investigation of low-Reynolds number propeller for different design parameters,” *AIAA Aviation Forum*, 2025.
- [36] “Data Interpolation with Radial Basis Functions (RBFs),” <https://shihchinw.github.io/2018/10/data-interpolation-with-radial-basis-functions-rbfs.html>, 2025. Accessed: 23 Jun 2025.
- [37] Cavoretto, R., De Rossi, A., and Perracchione, E., “Partition of unity interpolation on multivariate convex domains,” *International Journal of Modeling, Simulation, and Scientific Computing*, Vol. 6, No. 04, 2015, p. 1550034.
- [38] Shubham, S., Spinelli, A., and Kipouros, T., “Visual Analytics Framework for Multi-Disciplinary Multi-Objective Optimisation of Aircraft Design,” *Available at SSRN 5114731*, 2025.



DSPACE

<https://dspace.org/>

A multi-dimensional machine learning framework for superior propeller design choices

Shubham, Shubham; Sharma, Ankit

2025-07-29

American Institute of Aeronautics and Astronautics (AIAA)

<https://doi.org/10.2514/6.2025-3423>

Shubham S, Sharma A. (2025) A multi-dimensional machine learning framework for superior propeller design choices. In: AIAA Aviation Forum and Ascend 2025, 21-25 July 2025, Las Vegas, USA, Article number 2025-3423