A machine learning approach for propeller design and optimization

Part I

Doijode, Pranav Sumanth; Hickel, Stefan; van Terwisga, Tom; Visser, Klaas

DOI
10.1016/j.apor.2022.103178

Publication date
2022

Document Version
Final published version

Published in
Applied Ocean Research

Citation (APA)
Doijode, P. S., Hickel, S., van Terwisga, T., & Visser, K. (2022). A machine learning approach for propeller design and optimization: Part I. *Applied Ocean Research*, *124*, Article 103178. https://doi.org/10.1016/j.apor.2022.103178

Applied Ocean Research 124 (2022) 103178

Contents lists available at ScienceDirect

Applied Ocean Research

journal homepage: www.elsevier.com/locate/apor

A machine learning approach for propeller design and optimization: Part I

Pranav Sumanth Doijode [a,*], Stefan Hickel [b], Tom van Terwisga [a,c], Klaas Visser [a]

[a] Faculty of Mechanical, Maritime and Materials Engineering, Delft University of Technology, The Netherlands
[b] Faculty of Aerospace Engineering, Delft University of Technology, The Netherlands
[c] Maritime Research Institute Netherlands (MARIN), Wageningen, The Netherlands

ARTICLE INFO

Keywords:
Machine learning
Propeller design and optimization
Uncertainty
Dynamic optimization
Orthogonal parametric model

ABSTRACT

This paper introduces a machine learning approach for optimizing propellers. The method aims to improve the computational cost of optimization by reducing the number of evaluations required to find solutions. This is achieved by directing the search towards design clusters with good performance, i.e. high propulsive efficiency and low cavitation. Three types of clusters are expected. The first cluster constitutes designs with performance of interest, i.e. high efficiency and low cavitation. The second cluster constitutes designs with performance not of interest, i.e. low efficiency and high cavitation. The third cluster constitutes designs whose performance cannot be estimated with the Boundary Element Methods (BEM) that we use in this study. In simple cases with single objective optimization to maximize efficiency, these clusters can be identified a-priori with unsupervised classifiers provided that orthogonally independent parameters are used as demonstrated in this paper. For multi-objective constrained optimization, to maximize efficiency and minimize cavitation, for example, supervised classifiers may be required to learn the clusters. Classical design variables such as chordlength, pitch, skew, rake, thickness distribution and camber of hydrofoils cannot be used to identify these clusters because of multicollinearity. Thus, a new orthogonal parametric model is proposed where the parameters are directly derived from the propeller blade mesh. As the blade surface mesh is used as boundary conditions to solve the governing equations, the orthogonal parameters are expected to have a stronger correlation with performance predictions of BEM or Computational Fluid Dynamics (CFD) than classical design variables. We demonstrate that design clusters with good performance can be identified with few BEM evaluations. Furthermore, the method synergizes explainable supervised and unsupervised learning to advice search algorithms and quickly guide them to lucrative regions in the design space. However, reducing the cost of optimization results in a trade-off with completeness of the search; this is also investigated in this paper. The method is demonstrated on a simple fully wetted flow case of the benchmark Wageningen B-4 70 propeller with P/D=1.0, as the geometry and open-water curves are readily accessible allowing back of the envelope verification and validation of our results.

1. Introduction

The maritime industry has embraced its responsibilities towards the environment and is making active efforts to reduce Green House Gas (GHG) emissions. The International Maritime Organization (IMO) aims to achieve a 50% reduction in GHG emissions by 2050 compared to 2008 levels (IMO, 2019). Realizing this goal requires significantly improving the propulsive efficiency, in addition to improving thermodynamic efficiency, transitioning to cleaner fuels, and adopting better propulsion architectures. For modern vessels, propulsive efficiency is typically between 29%–35% (Theotokatos and Tzelepis, 2015). Improving the hydrodynamic efficiency of propellers is an important area of focus as it plays a critical role in determining the total propulsive efficiency. Optimizing propellers to improve efficiency and cavitation behaviour is computationally expensive even in the preliminary design phase. This is especially the case for clean-slate designs which do not use legacy propeller series. Traditional preliminary design of propellers involves two steps (Vesting, 2015): In the first step, systematic open-water studies of legacy propeller series are performed to identify the main design parameters. In the second step, a Design and Optimization (D&O) problem is solved to numerically determine specific design parameters. Currently, the maritime industry is transitioning to shape

* Corresponding author.
E-mail address: P.DoijodeSumanth@tudelft.nl (P.S. Doijode).

https://doi.org/10.1016/j.apor.2022.103178
Received 6 October 2021; Received in revised form 6 April 2022; Accepted 11 April 2022
Available online 9 May 2022

Nomenclature

$\bar{f}_*$	Posterior mean prediction for input x_*
ϵ	Independent and identically distributed noise
η	Open-water efficiency, $J.K_t/2\pi K_q$
γ	Skew angle in radians
λ	Rake angle in radians
$\mathbb{V}$	Posterior variance prediction
$\mathcal{N}(\mu, \sigma^2)$	Normal distribution
$\mathfrak{R}_{cart}$	A Cartesian frame of reference
$\mathfrak{R}_{cyl}$	A cylindrical frame of reference
$\mathfrak{R}_{orth}$	A reference frame defined by orthogonal vectors
$\mathbf{f}$	Latent function values, $\mathbf{f} = (f(x_1), \ldots, f(x_N))^T$
$\phi(\vec{x})$	Feature map of input $\vec{x}$
ρ_∞	Free-stream density
Σ	Diagonal matrix of singular values
σ_n^2	Noise Variance
E[X]	Expectation of any matrix X
K	$n \times n$ Gram matrix
k_*	The vector $\mathrm{K}(X, x_*)$
R[X, X]	Correlation coefficient matrix of any matrix X
θ	Pitch angle in radians
$\vec{\mu}, \vec{\mu^*}$	The mean vector for rows of M and M^*
$\vec{p}$	A point on a B-Spline surface
$\vec{r}_i$	Vector from origin to the mid-chord location
$\vec{s}$	A design vector of classical design variables
$\vec{x}$	The design vector of orthogonal parameters
C_f	Skin Friction Coefficient
D	Propeller diameter
f	Real valued function $f(x) : \mathbb{R}^N \to \mathbb{R}$
f_{obj}	Real valued objective function $f_{obj}(x) : \mathbb{R}^n \to \mathbb{R}$
g	A real valued constraint function $g(x) : \mathbb{R}^n \to \mathbb{R}$
J	Advance ratio, $v_\infty/\omega D$
k'	Number of orthonormal parameters
$k(x, x_*)$	Kernel function evaluation at inputs x and x_*
K_q	Torque Coefficient, $M_x/\rho_\infty v_{ref}^2 L_{ref}^3$
K_t	Thrust Coefficient, $T/\rho_\infty v_{ref}^2 L_{ref}^2$
M	Data Matrix of all propeller geometries in $\mathfrak{R}_{cyl}^{N' \times k}$
M^*	Parameter matrix for B-Spline discretization
M_x	Torque
N	Number of propellers in universal set
$N_{i,p}(t)$	A 3rd order pth degree basis function
p_{atm}	Absolute atmospheric pressure
S	The sample matrix of classical design variables
$s(\vec{x})$	The Silhouette Score of a vector $\vec{x}$
s_i	The ith classical design variable in $\vec{s}$
T	Thrust
t	Knots of the B-Spline surface
T_0	Target Thrust
U	Left singular vector matrix
u, v	B-Spline surface parameters
V	Right singular vector matrix
v_∞	Open-Water Speed
v_{cav}	Cavitation inception speed
v_{ship}	Ship speed
X	The parameter matrix
x_*	A new query vector
x_i	The ith orthogonal parameter in $\vec{x}$
y	A real valued range
y_c	Performance label for designs
k	Dimensionality

adaptive propellers (Liu and Young, 2009) and also aims for hydroacoustic optimization (Brizzolara and Gaggero, 2009). Designers may need to start from fresh propeller geometries whose open water curves are not available. Consequently, methods to numerically determine design parameters, i.e. D&O are starting to become the first step in preliminary design. Therefore, it is very important to the improve the reliability of D&O and also reduce its computational cost.

Optimization methods primarily focus on improving efficiency and cavitation behaviour. In addition to efficiency and cavitation objectives, propeller optimization must also account for constraints on hull excitation (Bosschers, 2009), cavitation erosion (van Terwisga et al., 2007), ship speed, thrust, and strength. Typically, the lower computational cost of Boundary Element Methods (BEM) motivates their use in the preliminary design phase to predict the efficiency and cavitation behaviour of a blade geometry. To generate the blade geometry, most methods rely on (a) classical design variables — pitch, diameter, hydrofoil camber, thickness distribution, skew and rake (Arapakopoulos et al., 2019; Esmailian et al., 2017), (b) control points of splines or coefficients of equations that define propeller families (Vesting et al., 2013; Gaggero et al., 2017; Bertetta et al., 2012) and, (c) free-form control points (Garg et al., 2017). Preliminary design methods typically use classical design variables as their values also communicate design specifications. However, classical design variables do not always lead to a good formulation of the optimization problem. An optimization problem is well formulated if design variables can be considered orthogonally independent. In the case of classical design variables, changing a single design variable perturbs multiple nodes on the mesh. Since, the surface is often defined with splines, the translation in mesh nodes is strongly correlated and also predictable. Furthermore, in most cases, multiple classical design variables tend to change the same mesh nodes. This results in *multicollinearity*, i.e. two or more variables being strongly correlated (Allen, 1997). Thus, it is difficult to isolate the impact of individual design variables on objectives and constraints.

Consider the objective

$$\max f = \eta\left(\vec{M}(\vec{s})\right), \tag{1}$$

where the blade geometry is defined by classical design variables $\vec{s}$, η is the hydrodynamic efficiency and $\vec{M}$ is the generated mesh for a blade geometry (see Eq. (8)). The optimization problem is well formulated if $\vec{s}$ can be considered orthogonally independent. The Jacobian of f by chain rule is

$$\boldsymbol{J}_{\mathbf{f}} = J_1 \cdot J_2 = \begin{bmatrix} \frac{\partial f}{\partial m_0} & \cdots & \frac{\partial f}{\partial m_q} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial m_0}{\partial s_0} & \cdots & \frac{\partial m_0}{\partial s_r} \\ \vdots & \ddots & \vdots \\ \frac{\partial m_q}{\partial s_0} & \cdots & \frac{\partial m_q}{\partial s_r} \end{bmatrix}, \tag{2}$$

where J_1 is the Jacobian of f with respect to the mesh $\vec{M} \in \mathfrak{R}^q$ and J_2 is the Jacobian of the mesh with respect to the design vector $\vec{s} \in \mathfrak{R}^r$. $\vec{s}$ can be considered orthogonally independent if J_2 is an orthonormal matrix, i.e. $\partial\vec{M}/\partial s_i \cdot \partial\vec{M}/\partial s_j = 0$. Because changing a single classical variable

perturbs multiple nodes of the mesh and multiple classical design variables tend to change the same mesh nodes, $\partial\vec{M}/\partial s_i \cdot \partial\vec{M}/\partial s_j \neq 0$. Thus, $\vec{s}$ cannot be considered orthogonally independent. This is also the case for other objectives/constraints such as suction side cavity volume ($V_{cav,b}$), pressure side cavity volume ($V_{cav,f}$) and tip-vortex cavity radius for a propeller blade. Hence, the relation between η, $V_{cav,b}$, $V_{cav,f}$, r_c and classical design variables is obscured.

Often, in the preliminary design phases, the use of Boundary Element Method (BEM) results in relatively low computational cost but comes with high uncertainties (Vaz et al., 2015). Thus, the relation between objectives, constraints and mesh (J_1 in Eq. (2)) is also obscured to an extent. Hence, optimization problems depending on classical design variables and BEM may require more evaluations than necessary to identify lucrative regions in the design space. This is irrespective of whether a gradient descent or gradient free search algorithm is used.

In this paper, we propose to reduce the computational cost of optimization by (a) addressing the challenge of multicollinearity which affects J_2 in Eq. (2), with a set of orthogonal parameters $\vec{x}$ such that $\partial\vec{M}/\partial x_i \cdot \partial\vec{M}/\partial x_j = 0$ and, (b) applying explainable machine learning to model the uncertainties in predictions made by BEM, which affects J_1 (Eq. (2)) in the preliminary design phase.

The orthogonal parameters are derived from Singular Value Decomposition (SVD) of a data matrix populated with variations of the blade geometry. There are two comparable applications of SVD. The first is for the symbolic reformulation of design and optimization problems (Sarkar et al., 2009) and the second is for the dimension reduction in single- and multi-disciplinary hull form optimizations (Diez et al., 2016). In both cases, relatively small data sets with large dimensionality are reduced to very few dimensions. In Sarkar et al. (2009), optimization problems are even clustered based on symbolic similarities.

BEM typically tends to predict similar performance, i.e. efficiency and cavitation behaviour, for comparable blade geometries. Thus, three clusters are expected during design and optimization. The first cluster constitutes of designs with performance of interest, i.e. high efficiency and low cavitation. The second cluster constitutes of designs with performance not of interest, i.e. low efficiency and high cavitation. The third cluster consists of designs whose performance cannot be estimated accurately due to limitations of the BEM used in this study. We consider the latter two clusters as designs not of interest. Thus, designs of interest are assigned the label $y_c = 1$ and designs not of interest are assigned the label $y_c = 0$. With the use of the proposed orthogonal parameters, it is expected that these clusters are discernable as the relation between hydrodynamic efficiency (η), $V_{cav,b}$, $V_{cav,f}$ and r_c are not obscured. In simple cases with single objective optimization, these clusters can be identified a-priori with unsupervised classifiers as demonstrated in this paper. Thus, fewer evaluations are required to identify lucrative regions in the design space. For multi-objective constrained optimization, supervised classifiers are required to learn the clusters and focus search in lucrative regions. The multi-objective constrained optimization with supervised classifiers is demonstrated in a separate publication (Doijode et al., 2022). Both papers use explainable classifiers (Abedjan et al., 2019) as they provide insight into *why* a prediction is made.

The resulting optimization method – *Dynamic Optimization* – is elaborated in Section 2. We call it Dynamic Optimization (DO) because the parameters are derived from the blade meshes that become boundary conditions for the equations governing the *dynamics* of the flow. Sections 2.1–2.5 detail the building blocks of the method. Section 3 details the demonstration setup and modelling of uncertainty. Section 4 presents the results on reliability and performance of the orthogonal parameters, supervised and unsupervised classifiers. The trade-offs between efficiency, completeness and optimality for Dynamic Optimization is demonstrated in Section 4.3. The conclusions are provided in Section 5.

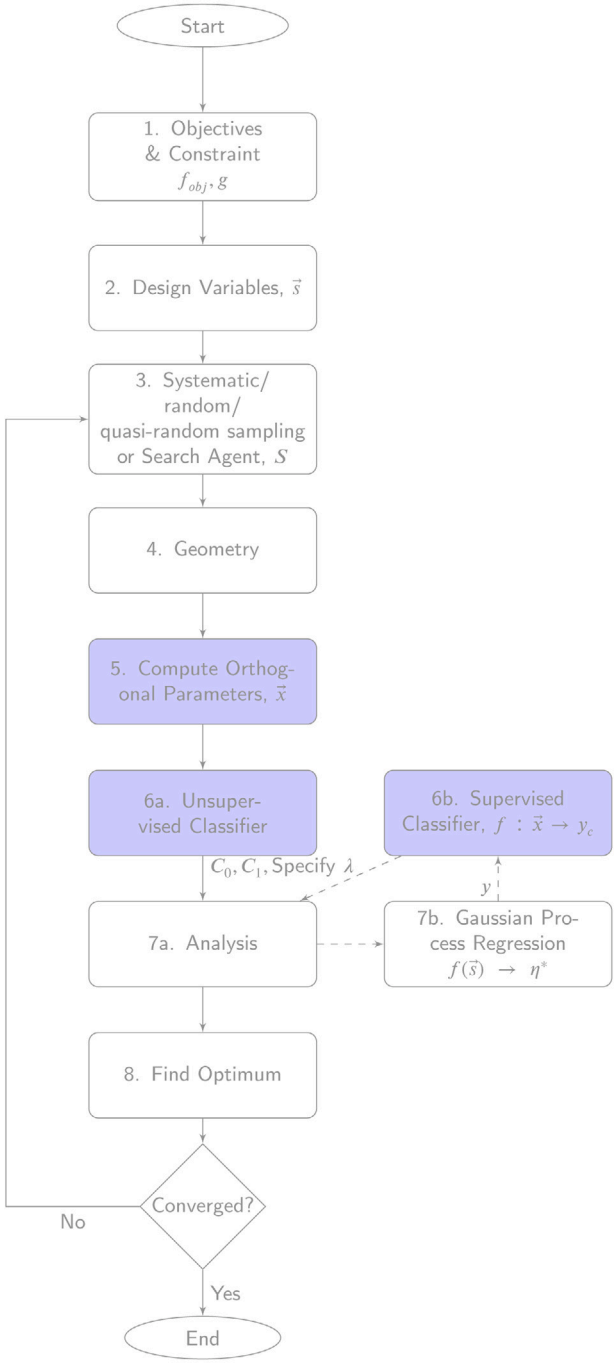

Fig. 1. Dynamic Optimization workflow.

2. Dynamic Optimization

Dynamic Optimization is a constraint respecting design method, where search agents are advised by classifiers. The workflow is described in Fig. 1 and the implementation of DO is detailed in Algorithm 1. In the figure each element of the method is labelled. At the start, objectives, constraints and design variables are specified (Block 1–2, Fig. 1). Then a search/sampling strategy based on systematic, random or pseudo-random sampling or another search agent is chosen (Block 3). Based on the sampling, the geometry is generated (Block 4, Fig. 1). After modelling, a set of orthogonal parameters ($\vec{x}^*$) that define the design are computed. These orthogonal parameters are projections of the propeller mesh $\vec{M}^*$ in a k' dimensional hyperspace ($\mathfrak{R}_{ortho}$).

Algorithm 1 Dynamic Optimization — Single Objective

Input: $f_1,\ g_1, g_2, ..., g_m, \lambda, N, \sigma^2$
Output: $\eta_{max}, \vec{s}_{sol}$

procedure DYNAMIC OPTIMIZATION - SINGLE OBJECTIVE
  $S, X \leftarrow \emptyset$
  $\eta_{max} \leftarrow 0$
  $S \leftarrow$ SEARCH(*params)
  for $i \leftarrow (0 \rightarrow N)$ **do**
    $\vec{s} \leftarrow S_i$
    Generate geometry and mesh ($\vec{M}*$) for $\vec{s}$
    $\vec{x}^* \leftarrow$ PROJECTION($\Re_{ortho}, k', \vec{M}*$) ▷ See Algorithm 4
    $X \leftarrow X \cup \vec{x}^*$
  $X \leftarrow$ NORMALIZE(X)
  $C0, C1 \leftarrow GET\_CLUSTERS(X, \lambda)$ ▷ See Algorithm 2, $\lambda \in (0, 100.0]$
  $C_{first}, C_{second}, \eta_{max}, \vec{s}_{sol} \leftarrow BEST\_CLUSTER(C0, C1)$ ▷ See Algorithm 3
  #At this point $\lambda/100 \times N$ designs have been evaluated and the better cluster is known.
  $X_{nb} \leftarrow \{x_i | i \in C0 \cup C1\}$
  GNB $\leftarrow$ GAUSSIAN-NB(X_{nb}, y_c) ▷ $y_c = 1$ for C_{first} and $y_c = 1$ for C_{second}
  $X' \leftarrow X - X_{nb}$
  if $S' \neq \emptyset$ **then**
    for $i \leftarrow (0 \rightarrow len(X'))$ **do**
      if GNB(x_i) == 1 **then**
        $\eta* \leftarrow$ EVAL(s_i) ▷ BEM or CFD.
        if $\eta* > \eta_{max}$ **then**
          $\eta_{max}, \vec{s}_{sol} \leftarrow \eta*, s_i$
      else
        Pass ▷ Skipped an evaluation and reduced cost of optimization.
  return $\eta_{max}, \vec{s}_{sol}$

Algorithm 2 Clustering Algorithm

Input: X, λ
Output: $L0, L1$

procedure GET_CLUSTERS
  $unsup\_clf \leftarrow$ GaussianMixture(n_clusters=2, initialization=kmeans)
  unsup_clf.train(X)
  $y_{c,unsup} \leftarrow unsup\_clf.predict(X)$
  $l1, l2 \leftarrow \emptyset$
  for $i \leftarrow (0 \rightarrow len(X))$ **do**
    if $y_{c,unsup} == 0$ **then**
      $l1 \leftarrow l1 \cup i$
    else
      $l2 \leftarrow l2 \cup i$
  $L0 \leftarrow$ RandomChoiseWithoutReplacement($l1, int(len(l1) * \lambda/100)$) ▷ Getting Indices between 0 and N
  $L1 \leftarrow$ RandomChoiseWithoutReplacement($l2, int(len(l2) * \lambda/100)$)
  return $L0, L1$

Section 2.4 details the derivation of orthogonal parameters. These parameters are defined to address the limitations of traditional parametric models. In the first step, based on the normalized orthogonal parameters (X), the unsupervised classifier finds two clusters ($C0, C1$) within which features are comparable (Block 6a, Fig. 1). It is also possible to increase the number of clusters; the number of clusters can be determined by the number of objectives, constraints and also the expected variance in the propeller geometry. Next, the designer needs to specify the percentage of designs (λ) to sample from each each cluster C_i for which the detailed analysis is performed with BEM or CFD (Block 7a, Fig. 1). The results of analysis indicates the cluster with best designs (C_{first}). C_{first} contains the solution ($\vec{s}_{sol}$) with the best efficiency (η_{max}). At this point λ percentage of designs have been evaluated and the best design is known. All the evaluated designs X_{nb} and corresponding cluster labels y_c are used to train the supervised explainable Naive-Bayes classifiers (GNB, Algorithm 1:L15 and Block 6b, Fig. 1). Based on learnt labels, the classifiers guide search agents to lucrative clusters in the design space while passing on designs that are not in the better cluster (Block 8, Fig. 1). False positives are caught with evaluations. The effect of false negatives can be controlled by increasing λ. Thus, the trade-off with completeness of optimization is controlled by the designer via the parameter λ. Finally, if all convergence criteria are satisfied, the design method reports the optimum else it is proposed to explore new regions of the design space. The subsequent sections detail the critical blocks of DO. For the method to be successful, the reliability of Blocks 5, 6a and 6b (emphasized in Fig. 1) are of critical importance.

2.1. Objective, constraints and design variables

Propeller design and optimization is a complex problem which requires numerous considerations. One of the primary objectives is to maximize efficiency (η). The problem is constrained by target thrust T_0 at a specified ship speed (v_{ship}). Other constraints to consider are

Algorithm 3 Best Cluster

Input: $C0, C1$
Output: $C\_first, C\_second, \eta_{max}, \vec{s}_{sol}$

procedure BEST_CLUSTER
  $\vec{\eta}_0, \vec{\eta}_1 \leftarrow \vec{0}_{len(C0)}, \vec{0}_{len(C1)}$
  $\eta_{max} \leftarrow 0.0$
  $\eta_{tr}, S_{tr} \leftarrow \emptyset$
  for $i \leftarrow \vec{\eta}_{0,0} \ldots . \vec{\eta}_{0,len(C0)}$ **do**
    $\eta_{0,i} \leftarrow \text{EVAL}(s_i)$
    $\eta_{tr} \leftarrow \eta_{tr} \cup \eta_{0,i}$
    $S_{tr} \leftarrow S_{tr} \cup s_i$
  for $i \leftarrow \vec{\eta}_{1,0} \ldots . \vec{\eta}_{1,len(C1)}$ **do**
    $\eta_{1,i} \leftarrow \text{EVAL}(s_i)$
    $\eta_{tr} \leftarrow \eta_{tr} \cup \eta_{1,i}$
    $S_{tr} \leftarrow S_{tr} \cup s_i$
  $\bar{\eta}^*, \mathbb{V}[\eta *] \leftarrow \text{GPR}(X = S_{tr}, y = \eta_{tr}, var = \sigma^2)$ ▷ See Eqs. (28a)–(28b)
  if max $\bar{\eta}$ **in** $C0$ **then**
    $C\_first, C\_second \leftarrow C0, C1$
  else
    $C\_first, C\_second \leftarrow C1, C0$
  $\eta_{max} \leftarrow \max \eta_{tr}$
  return $C\_first, C\_second, \eta_{max}, \vec{s}_{sol}$ ▷ $\vec{s}_{sol}$ corresponds to design with maximum efficiency.

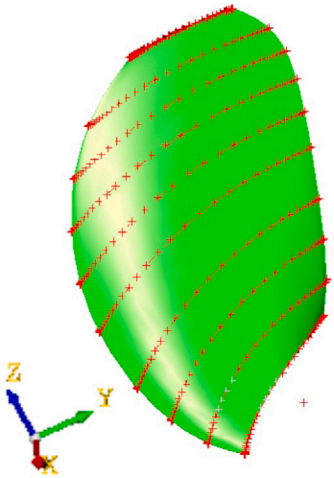

Fig. 2. Hydrofoil cross-sections and their locations.

cavitation, noise, hull induced pressure fluctuations and structural integrity.

In this demonstration, the single objective is to maximize the open-water hydrodynamic efficiency (η) of a Wageningen B4-70 propeller blade. Our starting geometry is a blade with P/D = 1.0. The propeller is modelled with 10 hydrofoil cross-sections as illustrated in Fig. 2. The constraints are (a) target thrust $T_0 = 27$ kN and (b) constant $v_{ship} = 1.72$ m s^{-1}. From the B4-70 series charts, we can find that B4-70 P/D = 0.8 is a better starting point for the chosen operating point. However, we choose our baseline to be B4-70 P/D = 1.0 which is further away from a known local optimum or known optimal search direction. This increases the chances of finding new search directions. In our optimization, the thrust is maintained by modifying the rotation rate of the propeller. The geometry is varied by modifying the pitch distribution, hence, the mean pitch also changes, see Fig. 3. Therefore, the pitch distribution is not restricted to that of the Wageningen B4 series. The baseline performance is listed in Table 3.

There are four primary reasons for defining a simplified optimization problem. Firstly, the simplicity of the problem enables a clear demonstration and better testing of each component of DO. In order to be applied to realistic problems (Part-II), DO must first work

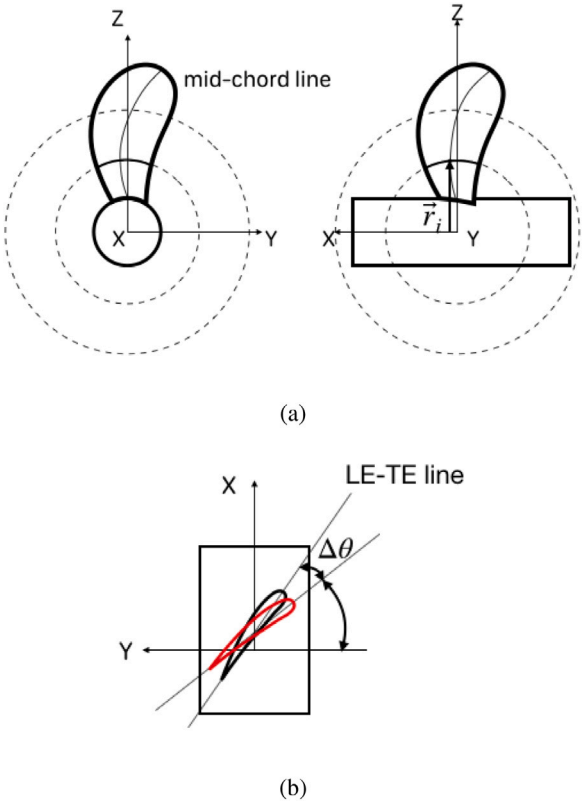

Fig. 3. (a) $\vec{r}_i$ and, (b) schematic illustrating pitch modification.

for simple propeller optimization problems. Secondly, the open-water curves of the propeller are thoroughly investigated and also publicly available (Barnitsas et al., 1981). Thus, the results of the current optimization can be quickly verified and validated with existing open-water curves. Thirdly, the geometry has been published and is readily available without knowledge embargo. Finally, the analysis tool PROCAL (Vaz and Bosschers, 2006) has been validated for the open-water fully wetted flow case of this propeller.

Classical design variables such as diameter (D), pitch distribution and camber correlate with η. A larger propeller tends to be more efficient and the upper-bound for diameter is often determined by hull-integration, tip-speed, hull induced pressure fluctuation and cavitation constraints. At the maximum possible diameter, it is possible to improve efficiency by modifying the pitch distribution. Cavitation behaviour is improved by modifying camber. As the primary objective of the demonstration case is efficiency, the aim is to find the optimal pitch distribution.

To modify the blade pitch, we use the parameter θ. Fig. 3(b) illustrates the change in geometry when θ is varied. θ is modified from the 4th–10th cross-sections i.e. mid section to blade tip. The target cross-section is rotated by $\Delta\theta$ about the $\vec{r}_i$. $\vec{r}_{\vec{i}}$ is the vector parallel to z-axis which points to the midpoint of the LE-TE line as illustrated in Fig. 3(a). Pitch is expected to be sensitive to $\Delta\theta$. An increase in θ results in a higher pitch exposing the blade to higher inflow angles thus also higher loading. Note that $\Delta\theta < 0$ implies that the cross-section was depitched. Otherwise the cross-section's pitch either stayed the same or it increased. While θ may influence skew (γ) and rake (τ), they are not expected to be very sensitive to θ.

2.2. Halton sampling

To demonstrate the existence of clusters, the Halton sequence (Halton, 1960) is used to generate samples. The sequence is chosen as it is a systematic, repeatable and quasi random sampling method. A possible strategy to improve hydrodynamic efficiency is to reduce rotational losses by off-loading the blade (Schuiling and van Terwisga, 2016). Thus, pitches from the 4th–10th cross-sections are chosen as design variables. They are perturbed from the baseline design by $\Delta\theta_{p,i} = \pm 0.150$.

Within the stated bounds, 200 points are sampled with the Halton sequence to generate the sample data matrix

$$S = \begin{bmatrix} \Delta\theta_{1,4} & \cdots & \Delta\theta_{1,10} \\ & \vdots & \\ \Delta\theta_{N,4} & \cdots & \Delta\theta_{N,10} \end{bmatrix}. \tag{3}$$

The sequence is characterized by low-discrepancy and low-dispersion compared to random sampling methods when $k < 10$, where k is dimensionality of the sampling space (LaValle, 2006). The modified geometries are constructed with B-Splines as detailed in the following section.

2.3. Geometry

The blade geometry is defined by a single B-Spline surface (Prautzsch et al., 2002) defined as

$$\vec{p}(u,v) = \sum_{i=0}^{m}\sum_{j=0}^{n} N_{i,d_1}(u) N_{j,d_2}(v) \vec{P}_{i,j}, \tag{4}$$

where $u, v \in [0.0, 1.0]$ are knots in the u and v directions; $\vec{P}_{i,j}$ is the control point of the ith row and jth column; $m+1$ and $n+1$ are the number of rows and columns the control points are organized in; d_1, d_2 are the degree of the B-Spline in u and v directions respectively and; N_{i,d_1} and N_{j,d_2} are the basis function in u and v directions respectively. The surface is fitted to hydrofoil cross-sections in Cartesian coordinates at different radial locations. In our case $d_1 = 3$ as the spline is closed in u direction and $d_2 = 8$ to find an optimal fit for a hydrofoil with camber and thickness distribution with fewer control points. The basis functions ($N(t)$) for each directions are defined as

$$N_{i,0}(t) = \begin{cases} 1; t_i \le t < t_{i+1} \\ 0; otherwise, \end{cases} \tag{5}$$

$$N_{i,d} = \frac{t - t_i}{t_{i+d} - t_i} N_{i,d-1}(t) + \frac{t_{i+d+1} - t}{t_{i+d+1} - t_{i+1}} N_{i+1,d-1}(t). \tag{6}$$

where t is the knot value and d is the degree of the spline. Based on the geometry, the orthogonal parameters are derived.

2.4. Orthogonal parametric model

The primary objective of the proposed parametric model is to alleviate the limitations of classical design variables. Thus, the parametric model must (a) accurately quantify geometric variations, (b) yield orthogonally independent parameters, i.e. $\frac{\partial \vec{M}}{\partial x_i} \cdot \frac{\partial \vec{M}}{\partial x_j} = 0$ and, (c) be generalized for all possible geometries. To satisfy the above requirements, the parameterization is based on the coordinates of the blades' surface mesh such that all possible variations of all possible families are represented and the relation between geometry and design objectives is not obscured. The orthogonal parameters $x_i \in \vec{x}$ of the proposed parametric model are derived by projecting the surface mesh of propellers in an orthonormal hyperspace $\Re_{orth}$ to ensure that the parameters are orthogonally independent. Defining $\Re_{orth}$ involves four steps.

The *first step* is to ensure that the mesh used for the analysis of each design is comparable. Thus, the surface mesh is generated by uniformly discretizing the B-Spline surface in the B-Spline parametric space, i.e. along the u and v directions of the spline surface. The mesh M^* consisting of discrete nodes in the B-Spline parametric space is defined in Eq. (7), where m is the number of nodes on the hydrofoil cross-section and n is the number of cross-sections along the diameter. Thus, the mesh for the ith propeller $\vec{M}_i \in \Re^k_{cart}$ with dimensionality $k = 3 \times m \times n$ is

$$M^*_{m\times n} = \begin{bmatrix} (0.0, 0.0) & \cdots & \left(0.0, \frac{(j-1)}{(n-1)}\right) & \cdots & (0.0, 1.0) \\ \vdots & \ddots & & & \vdots \\ \left(\frac{(i-1)}{(m-1)}, 0.0\right) & & \left(\frac{(i-1)}{(m-1)}, \frac{(j-1)}{(n-1)}\right) & & \vdots \\ \vdots & & & \ddots & \vdots \\ (1.0, 0.0) & \cdots & \cdots & \cdots & (1.0, 1.0) \end{bmatrix}, \tag{7}$$

$$\vec{M}_i = \{p(m^*_{ij}) | m^*_{ij} \in M^*_{m\times n}\}. \tag{8}$$

The *second step* is to build the data matrix M by stacking meshes of all N sampled propellers and then transforming them into cylindrical coordinates:

$$\tilde{M} = \begin{bmatrix} \vec{M}_1 \\ \vdots \\ \vec{M}_N \end{bmatrix}, \tag{9}$$

$$M = T(\tilde{M}), T : \Re^{N\times k}_{cart} \rightarrow \Re^{N\times k}_{cyl}. \tag{10}$$

This pre-processing step of stacking and transforming the meshes into cylindrical coordinates reduces the variance in $M_{(:,i)} \forall i \in \{0, 3, 6...\}$ as cross-sections are defined at a constant radius from the propeller axis. Thus, variance in form and shape of the blade is emphasized in the data matrix.

The *third step* is to define the origin of $\Re_{orth}$. The origin is chosen to be the mean

$$\vec{\mu} = \frac{\sum_{i=1}^{N} \vec{M}_i}{N}, \tag{11}$$

of all N propeller meshes, leading to the translated data matrix

$$M' = \begin{bmatrix} \vec{M}_1 - \vec{\mu} \\ \vdots \\ \vec{M}_N - \vec{\mu} \end{bmatrix}. \tag{12}$$

The *fourth step* is to factorize M' using thin SVD (Van Loan and Golub, 1996), that is,

$$M'_{k\times N} = U_{k\times N} \cdot \Sigma_{N\times N} \cdot V'_{N\times N}. \tag{13}$$

This yields the left singular vectors U, the singular values Σ and the right singular vectors V. The product of the k' left and k' right singular vectors yields the orthonormal reference frame

$$\mathfrak{R}_{orth} = U_{k\times k'} \cdot V'_{k'\times k'}. \tag{14}$$

$\mathfrak{R}_{orth}$ is defined by k' orthonormal vectors and any surface mesh $\vec{M}$ can now be projected in $\mathfrak{R}_{orth}$. Thus the surface mesh can be represented as

$$\vec{M} = \sum_{i=0}^{k'} M cos(x_i)\hat{\mathfrak{R}}_{orth:,i} \tag{15}$$

where $M = |\vec{M}|$, $\hat{\mathfrak{R}}_{orth:,i}$ is the ith unit basis vector of the reference frame $\mathfrak{R}_{orth}$ and x_i is the projection angle of the mesh on the ith unit basis vector. Thus, the orthogonal parameters we propose are x_i. The orthogonal parameters can be represented by $\vec{x} \in \mathfrak{R}^{k'}$. Note that $\frac{\partial \vec{M}}{\partial x_i} \cdot \frac{\partial \vec{M}}{\partial x_j} = 0$ for all values of k'.

The procedure to project the mesh in $\mathfrak{R}_{ortho}$ and compute the orthogonal parameters is detailed in Algorithm 4. This algorithm then yields the Parameter Matrix (X) with values for each vector $\vec{x}$. As $k' \ll k$, it is possible to significantly reduce the number of parameters while not obscuring the relation between the geometry and the mesh, thus also the solution. Hence, it is ensured that this parameterization also opens doors for dimension reduction.

One limitation of the proposed parametric model is that perturbing orthogonal parameters results in non-conformal projections in the orthonormal spaces. These projection are non-conformal because they may not result in smooth splines which are required for propeller blade geometries. Thus, models similar to conformal deformation proposed for Free Form Deformation (FFD) (Peri, 2012) need to be formulated for orthogonal parameters as well. Another possibility is to establish a one-to-one map between orthogonal parameters and classical design variables to allow for conformal modifications of the geometry. In the current demonstration, we use classical design variables to modify the geometry and orthogonal parameters as features for classifiers and regressors.

Algorithm 4 Subspace Projection

procedure PROJECTION
  $M_{N\times k} \leftarrow$ Data matrix of designs to analyze
  Compute $\mathfrak{R}_{ortho}$
  $\mathfrak{R} \leftarrow \mathfrak{R}_{ortho}$
  $X \leftarrow 0_{N,k'+1}$
  for $i = 1$ to N **do**
    for $j = 1$ to N' **do**
      $X_{i,j} \leftarrow cos^{-1}\left(\frac{M_{i,:}\cdot\mathfrak{R}_{:,j}}{|M_{i,:}||\mathfrak{R}_{:,j}|}\right)$
  return $X_{N,k'+1}$

2.5. Classifiers

Both supervised and unsupervised classifiers are employed. Explainable machine learning models are opted for because they provide insights into why certain predictions are made. The goal of the unsupervised classifier is to identify existing clusters in the design space with comparable geometries and the goal of supervised classifiers is to learn which cluster of geometries are lucrative. Once trained, the supervised classifiers can guide search algorithms towards more lucrative regions in the design space thus reducing the number of required evaluations. In this paper, implementations of supervised and unsupervised classifiers in SciKit-Learn (Pedregosa et al., 2011) are used.

The classifiers use orthogonal parameters (X) as features given that they capture geometric variance reliably. In principle, the orthogonal parametric model is expected to satisfy the *i.i.d* assumption; thus, classifiers based on both generative models $p(X|y_c)$ and discriminate models $p(y_c|X)$ are applicable. In the following sections, we provide a concise description of the classifiers. For more details, the readers are advised to read the references in the respective subsections.

2.5.1. Unsupervised classifier

In the zero-th design iteration, since the number of clusters is predetermined and clusters are expected to differ in shape and share boundaries, Gaussian Mixture Models (GMM) with full covariance matrices (Bishop, 2006) is employed. The GMM classifier predicts class labels y_c as

$$y_c = \text{argmax}_j\ \pi_j p(\vec{x}_i|\mu_j, \sigma_j^2), \tag{16}$$

where j maximizes the probability of $\vec{x}_i$ occurring. In Eq. (16), $\pi_j = n_j/N$ is the ratio of the number of designs in the jth cluster (n_j) to the total number of designs N, thus it is also the cluster probability. The means μ_j and variance σ_j^2 for the jth cluster are found using the Expectation–Maximization algorithm with a Euclidean distance based K-Means centroid initialization.

2.5.2. Supervised classifiers

Naive-Bayes (NB) with kernel approximations of Support Vector Classifiers (SVC) (Chang and Lin, 2011) and Gaussian Process Classifiers (GPC) (Rasmussen and Williams, 2006) are compared. The classifiers are as follows:

Naive-Bayes

The Naive-Bayes classifier predicts labels as

$$y_c = \text{argmax}_j\ \pi_j p(\vec{x}_i|\vec{\mu}_j, \vec{\sigma^2}_j), \tag{17}$$

where j maximizes the probability of $\vec{x}_i$ occurring. In Eq. (17), the probability (p)s of $\vec{x}_i$ occurring given the mean and variance $\vec{\mu}_j, \vec{\sigma^2}_j$ for the jth cluster is

$$p(\vec{x}_i|\mu_j, \sigma_j^2) = \prod_{d=1}^{k'} \frac{1}{\sqrt{2\pi\sigma_{j,d}^2}}.\exp\left(-\frac{|x_{i,d} - \mu_{j,d}|^2}{2\sigma_{j,d}^2}\right). \tag{18}$$

k' is the number of orthogonal parameters, $x_{i,d}$ is the dth parameter in $\vec{x}_i$, $\mu_{j,d}$ and $\sigma^2{}_{j,d}$ are the mean and variance of $X_{:,d}$.

Support vector classifier

The decision function for SVC predicts the label y_c as

$$y_c = sgn\left(\sum_{n=1}^{N} y_i\alpha_i \mathrm{k}_* + \rho\right), \tag{19}$$

where $y_i\alpha_i$ are the dual coefficients, k_* is the correlation vector and ρ is the intercept. For our comparison study, the ith components of the correlation vector are computed with the Radial Basis Function (RBF) kernel,

$$k_i(x_i, x_*) = \exp(-|x_i - x_*|^2). \tag{20}$$

Multi-label classifications are implemented with an one-against-one approach (Knerr et al., 1990). In Eq. (19), the dual coefficients ($y_i\alpha_i$) are solutions to the dual problem of SVCs primal problem (Chang and Lin, 2011). The dual problem defined as

$$\min_\alpha \frac{1}{2}\alpha^T Q\alpha - \mathbb{1}^T\alpha, \tag{21a}$$

$$\text{subject to } y^T\alpha = 0; 0 \le \alpha_i \le C; \tag{21b}$$

where $C > 0$ is the upper bound, $Q_{ij} = y_i y_j \phi(x_i)\phi(x_j)$ and $\phi(x_i), \phi(x_j)$ are the feature map of input x_i, x_j.

Gaussian Process Classifier

The Gaussian Process Classifier (GPC) is based on the Laplace approximation method described by (Rasmussen and Williams, 2006). The posterior prediction ($\bar{\pi}_*$) for a query x_* is

$$\bar{\pi}_* = \int \sigma(f_*) p(f_*|X, y, x_*) d f_*. \tag{22}$$

Multi-label predictions can be made with both one-against-one and one-against-rest approaches.

In Eq. (22), the *first* term, $\sigma(f_*) = \sigma(f(\vec{x}_*)) = \sigma(x^T w)$. GPC places a GP prior with a RBF kernel, Eq. (20), on a latent function $f(\vec{x}) = x^T w$. The range of $f(\vec{x})$ is $(-\infty, +\infty)$. Mapping the range to $[0, 1]$ allows it to be interpreted as probability of labels in soft classification between two classes $y_c \in \{0, 1\}$. Thus the range of $f(\vec{x})$ is squashed through the logistic link function

$$\sigma(f(\vec{x})) = \frac{1}{1 + exp(-f(\vec{x}))}, \tag{23}$$

that maps the domain $(-\infty, +\infty)$ onto $[0, 1]$.

In Eq. (22), the *second* term $p(f_*|X, y, x_*)$ can be approximated with the Laplace approximation method. The term can be expanded as

$$p(f_*|X, y, x_*) = \int p(f_*|X, x_*, \mathbf{f}) p(\mathbf{f}|X, y) d\mathbf{f}, \tag{24}$$

where $p(\mathbf{f}|X, y) = p(y|\mathbf{f})p(\mathbf{f}|X)/p(\mathbf{y}|X)$. In the Laplace approximation method, a Gaussian approximation $q(\mathbf{f}|X, y)$ to the posterior of the latent function $p(\mathbf{f}|X, y)$ is utilized. The Gaussian approximation is obtained by doing a second order Taylor expansion of $\log(p(\mathbf{f}|X, y))$ around the maximum of the posterior as defined below:

$$q(\mathbf{f}|X, y) = \mathcal{N}(\mathbf{f}|\hat{\mathbf{f}}, A^{-1}) \approx exp(-\frac{1}{2}(\mathbf{f} - \hat{\mathbf{f}})^T A(\mathbf{f} - \hat{\mathbf{f}})), \tag{25}$$

where $\hat{\mathbf{f}} = \text{argmax}_{\mathbf{f}} p(\mathbf{f}|X, y)$ and $A = -\nabla\nabla log p(\mathbf{f}|X, y|(\mathbf{f} = \hat{\mathbf{f}}))$ is the Hessian of the negative log posterior at the point.

2.5.3. Classification labels (y_c)

Designs of interest are labelled $y_c = 1$ and designs not of interest are labelled $y_c = 0$. y_c is determined as

$$y_c = \begin{cases} 0 & \bar{\eta}_*(\vec{s}, T_0) \in [\eta_{min}, \eta_{min} + \delta] \\ 1 & \bar{\eta}_*(\vec{s}, T_0) \in [\eta_{min} + \delta, \eta_{max}]. \end{cases} \tag{26}$$

Note that the labels are based on posterior efficiency prediction $\bar{\eta}_*$ and a parameter $\delta = (\eta_{max} - \eta_{min})(2/3)$, where η_{max}, η_{min} are the maximum and minimum efficiencies of all designs in the samples. $y_c = 1$ indicates a lucrative cluster where search is to be focused.

2.5.4. Uncertainty in early design

The posterior efficiency prediction $\bar{\eta}_*$ is made by modelling the uncertainties in BEM predictions (Vaz et al., 2015). Typically, uncertainties are addressed in the detailed design stage. Those resulting from geometry and operational parameters are dealt with Robust Design methods (Park et al., 2006), where the primary objective is to reduce the parameter dependent variance in performance. Uncertainties resulting from modelling and discretization errors are dealt with higher-fidelity Computational Fluid Dynamics (CFD). However, delayed consideration of uncertainties results in lower marginal gains in performance. Thus, there is a need to account for uncertainties already in preliminary stages of design. For this purpose, Gaussian Process Regression is employed as detailed in this sub-section.

There are two primary sources of uncertainty. The first is the use of BEM for which performance predictions could have uncertainties as high as 15% (Vaz et al., 2015). The second is uncertainties resulting from regression itself. Both uncertainties are accounted for in a linear regression model with Gaussian noise as described by (Rasmussen and Williams, 2006):

$$y = f(s) + \epsilon, \tag{27a}$$

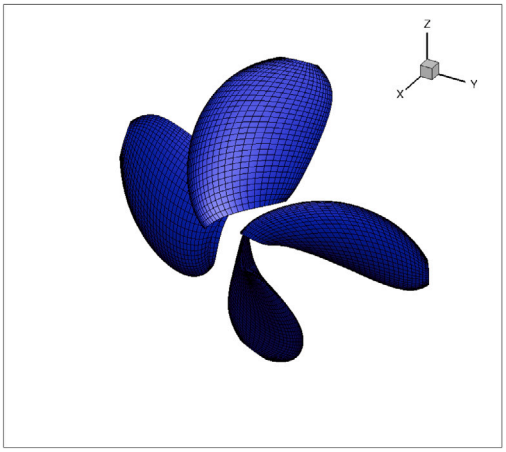

Fig. 4. Wageningen B4-70 propeller.

Table 1
Operational parameters.

Parameter	Units	Value
Open-Water Speed (v_∞)	m s^{-1}	[0.18–3.6]
Free-stream density (ρ_∞)	kg m^{-3}	1025
Absolute atmospheric pressure (p_{atm})	Pa	102 500
Propeller rotation speed (ω)	rev s^{-1}	15

$$f(s) = \phi(s)^T w, \tag{27b}$$

$$\epsilon \sim N(0, \sigma_n^2). \tag{27c}$$

Eq. (27a) describes the response y with noise ϵ. Note that ϵ models both analysis and regression uncertainties. The function $f(\vec{s}) : \mathbb{R}^N \to \mathbb{R}$ is assumed to be linear as described in Eq. (27b). The inputs $\vec{s} \in \mathbb{R}^N$ are projected onto feature spaces by the function ϕ. The vector space of the feature vector is the feature space. The noise ϵ is assumed to be an *i.i.d* Gaussian distribution with $\mu = 0$ and variance σ_n^2 as described in Eq. (27c) is used. The posterior prediction of mean and variance are

$$\bar{f}_* = \mathbf{k}_*^T (K + \sigma_n^2 I)^{-1} \mathbf{y}, \tag{28a}$$

$$\mathbb{V}[f_*] = k(s_*, s_*) - \mathbf{k}_*^T (K + \sigma_n^2 I)^{-1} \mathbf{k}_*. \tag{28b}$$

In Eqs. (28a)–(28b), K is the Gram matrix and $\mathbf{k}_*$ is the correlation vector. In this study, the components of the correlation vector and Gram matrix are computed with an RBF kernel, see Eq. (20).

3. Demonstration setup

As a benchmark case, the Wageningen B4-70 propeller with P/D = 1.0 is chosen. The geometry and mesh is illustrated in Fig. 4.

PROCAL is validated for the operating parameters listed in Table 1. The validation results are illustrated in Fig. 5. The experimental data (*Exp.*) in Fig. 5 is bundled and shipped with PROCAL. It can be observed that the PROCAL efficiency predictions match well with experimental observations up-to $J \approx 0.9$ and thereafter uncertainties increase. In this demonstration, the hub is not modelled. However, PROCAL predictions for hydrodynamic efficiency are comparable to experimental observations. This could be because of weak blade-hub interactions. One limitation of the current approach is that root cavitation is not captured during the optimization and if blade-hub interactions are strong the uncertainties in performance predictions could also be higher.

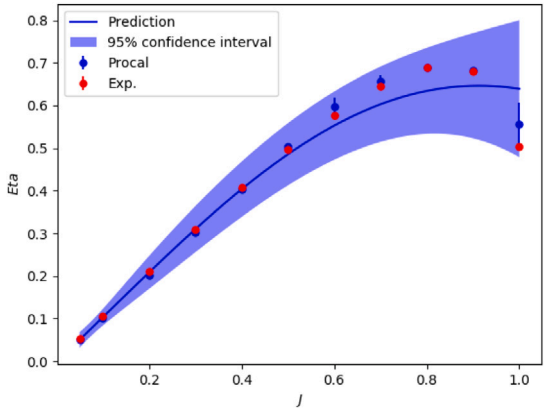

Fig. 5. PROCAL validation and Gaussian Process Regression results of open-water curve with modelled uncertainties.

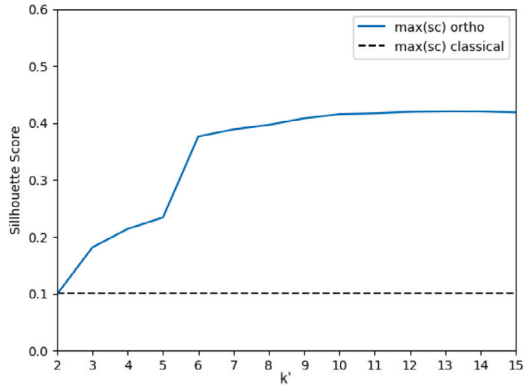

Fig. 6. Silhouette Scores for 2 parameter combinations with $k' \in [2, 15]$.

In the current setup, the resulting uncertainties in performance prediction are accounted for with the GPR estimator. The posterior efficiency prediction ($\bar{\eta}_*$) (-) and confidence interval is estimated with the Gaussian Process Regression (GPR) model defined in Eqs. (28a)–(28b). The variance is estimated to be 20% of the response η. This yields a 95% confidence interval that encapsulates predictions, experimental observations as well as uncertainties resulting from regression. For the GPR estimator, the mean prediction (-) would coincide with the experimental observations if the variance is assumed to be 0%. However, assuming a variance of 0% implies that PROCAL predictions do not differ from experimental observations, which is not the case. For varied geometries and operational conditions the residuals of the panel method can be different. Thus, the variance expected can be different from the estimated 20% in this demonstration. For the sake of simplicity, we assume the expected variance to be a constant. However, the trends in variance w.r.t geometries, operational conditions and residuals of numerical methods, if available, can also be considered in the GPR estimator during training (Block 7b in Fig. 1). This could lead to better cluster predictions made by the supervised classifier (Block 6b in Fig. 1). Better cluster predictions could reduce the number of iterations required to convergence.

4. Results

For DO to be successful, orthogonal parametric model (Block 5) and classifiers (Block 6a–6b in Fig. 1) need to be effective and reliable. The orthogonal parametric model is deemed reliable if it (a) is able to identify spatially discernible clusters and, (b) preserves the geometric correlations. The unsupervised classifiers are deemed reliable when they identify existing clusters. The supervised classifiers are deemed reliable if they direct search to lucrative clusters while continuously learning. The following sections report the results of reliability tests for the critical blocks and DO itself.

4.1. Orthogonal parametric model reliability

In this section, the results of the tests on reliability of the orthogonal parametric model are discussed. The ability of the orthogonal parametric model to yield spatially discernible clusters are determined with *silhouette scores*. The *silhouette score* (Rousseeuw, 1987) indicates whether there are spatially separated clusters in the data. The score for a design $\vec{x}_i$ is defined as

$$S(\vec{x}_i) = \frac{b(\vec{x}_i) - a(\vec{x}_i)}{max\{a(\vec{x}_i), b(\vec{x}_i)\}}, \tag{29a}$$

$$a(\vec{x}_i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, i \neq j} d(i, j), \tag{29b}$$

$$b(\vec{x}_i) = \min_{l \neq i} \frac{1}{|C_l|} \sum_{j \in C_l} d(i, j), \tag{29c}$$

where $a(\vec{x}_i)$ is the average dissimilarity of $\vec{x}_i$ with all other designs in the same cluster and $b(\vec{x}_i)$ is the minimum value of average dissimilarity with another cluster. This results in $S(\vec{x}_i) \in [-1, 1]$, with -1 indicating wrong labels, 0 indicating similarities between designs from different clusters thus, no clearly discernible clusters, and 1 indicating clearly distinguished clusters. Discerning clusters in a high-dimensional hyperspace is challenging. However, if clusters do exists, they would be discernible in combinations of any two parameters, $s_i, s_j \in \vec{s}$ or $x_i, x_j \in \vec{x}$. Thus, *silhouette scores* are calculated for all possible combinations of two parameters in $\vec{s}$ and $\vec{x}$.

The maximum score of all possible parameter pairs is a good indicator of performance and is plotted in Fig. 6. The orthogonal parameters tend to perform significantly better than classical design variables with 4 times higher silhouette scores. It is also observed from Fig. 6 that the score does not improve anymore for $k' > 11$. This indicates that most of the variance is captured in the first 10 orthonormal vectors defining $\mathfrak{R}_{orth}$. Thus, representing the designs in higher dimensions does not yield more distinguished clusters.

Fig. 7 illustrates the clusters generated by classical and orthogonal parameters for thrust requirement $T_0 = 27$ kN. Note that $s(1) = \Delta\theta_4$, thus in Fig. 7(a), $s(3), s(5)$ represent normalized $\Delta\theta$ for the 7th and 9th cross-section respectively. For classical design variables, it is observed in Fig. 7(a) that there are no clearly distinguishable design clusters. However, it is observed in Fig. 7(b) that orthogonal parameters result in marginally overlapped clusters with one of the two containing lucrative designs.

4.2. Classifier performance

In this section, the results of reliability tests for the classifiers (Block 6a–6b) are presented and discussed. The ability of the unsupervised classifier – Gaussian Mixture Model Classifier – to identify existing clusters with orthogonal parameters and classical design parameters is tested. Thereafter, the supervised classifiers' reliability in predicting whether a design is efficient or otherwise is tested with precision scores.

4.2.1. Gaussian Mixture Model (GMM)

The GMM classifier is used in the zero-th design iteration when no performance information is available yet. The classifier's ability to generalize depends strongly on the presence of discernible clusters. Testing the classifier's reliability via validation studies is not possible in principle as the ground truth is not known. Thus, the reliability is often implied from the posterior silhouette scores.

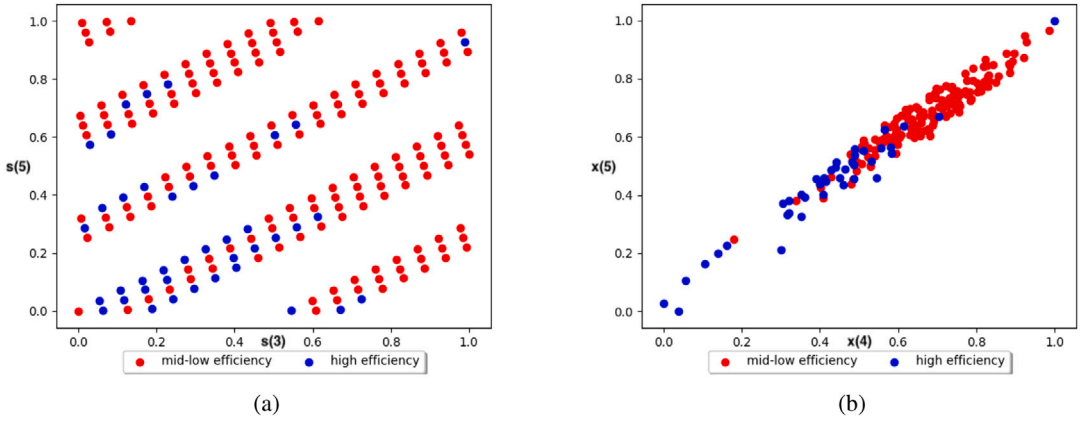

Fig. 7. Best spatial clustering parameter combinations at $T_0 = 27$ kN for (a) Classical design variables $\Delta\theta_7$ and $\Delta\theta_9$, $s(\{\Delta\theta_7, \Delta\theta_9\}) = 0.1007$. (b) Orthogonal design parameters x_4, x_5 when $k' = 13$, $s(\{x_4, x_5\}) = 0.4206$.

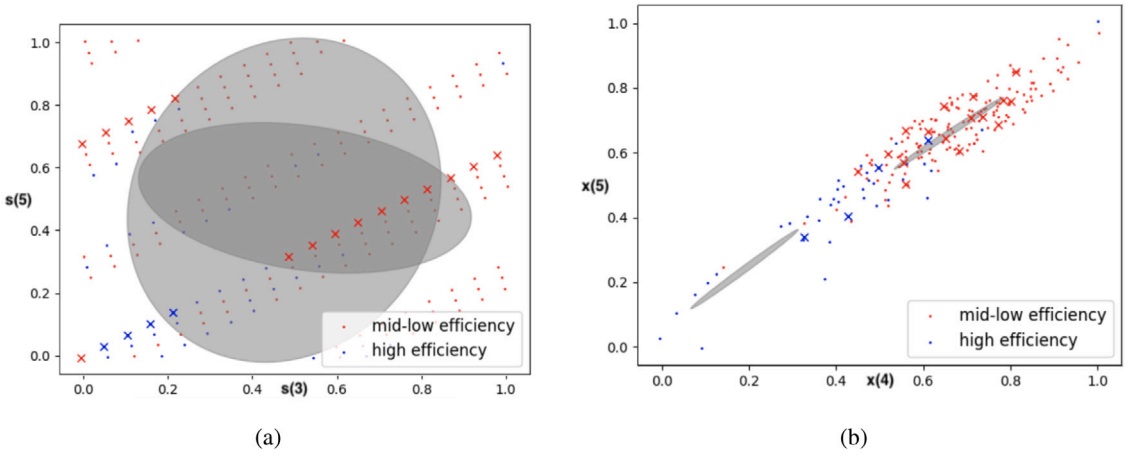

Fig. 8. GMM classifiers confidence ellipsoides for (a) Classical design variables $\{\Delta\theta_7, \Delta\theta_9\}$ and (b) Orthogonal parameters $\{x_4, x_5\}$, when $k' = 13$.

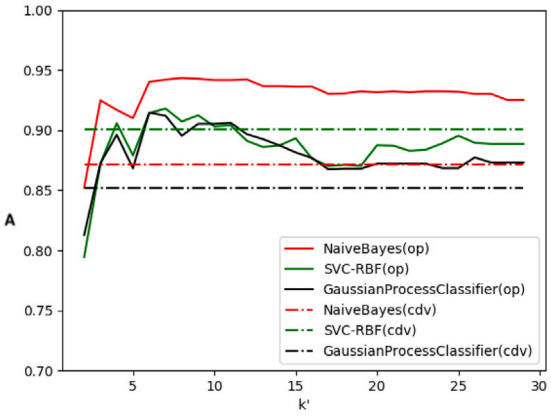

Fig. 9. Mean precision scores for stratified 10 fold cross validation studies of supervised classifiers.

From the silhouette scores (Fig. 6), it is observed that certain parameter combinations have high silhouette scores. Thus, for parameter combinations with high scores, confidence ellipsoids of the trained unsupervised GMM classifier are plotted in Figs. 8(a)–8(b) . It is observed that for classical parameters, the confidence ellipsoids of the classifier significantly overlap. This is clear indication that a-priori clustering of designs is not possible with classical parameters. On the contrary, there is no overlap in the confidence ellipsoids of orthogonal parameters indicating that a-priori clustering of designs is possible.

4.2.2. *Supervised classifiers*

The reliability of the NB classifier, SVC and GPC is evaluated with *precision scores*. The labels (y_c) for the training and testing samples are detailed in Section 2.5.3. The precision score for a classifier

$$A = \frac{tp}{tp + fp}, \tag{30}$$

is the ratio of true positive (tp) predictions to the sum of true positive and false positive (fp) predictions. This scoring is a reliable indicator when the number of members in each class for the testing set are disproportionate. Precision scores, $A > 0.5$ indicate that the classifier predictions had more true positives than false positives. The machine learning models used in DO aim to maximize true positives and the optimization method is designed to catch false positives thus improving reliability. Weighted average precision scores for all class labels are determined with 10 fold stratified cross validation studies. This reduces the risk of missing false negatives.

Fig. 9 illustrates the results of the cross validation study for the classifiers. The plots show the mean cross-validation precision score for predicting $y_c = 0$, i.e. for an inefficient design. It is observed that all classifiers have $p > 0.5$. Thus, true positive predictions are more likely than false positive predictions.

It is observed that precision scores for classifiers using orthogonal parameters are typically higher than those using classical design variables. However, SVC with orthogonal parameters tends to perform

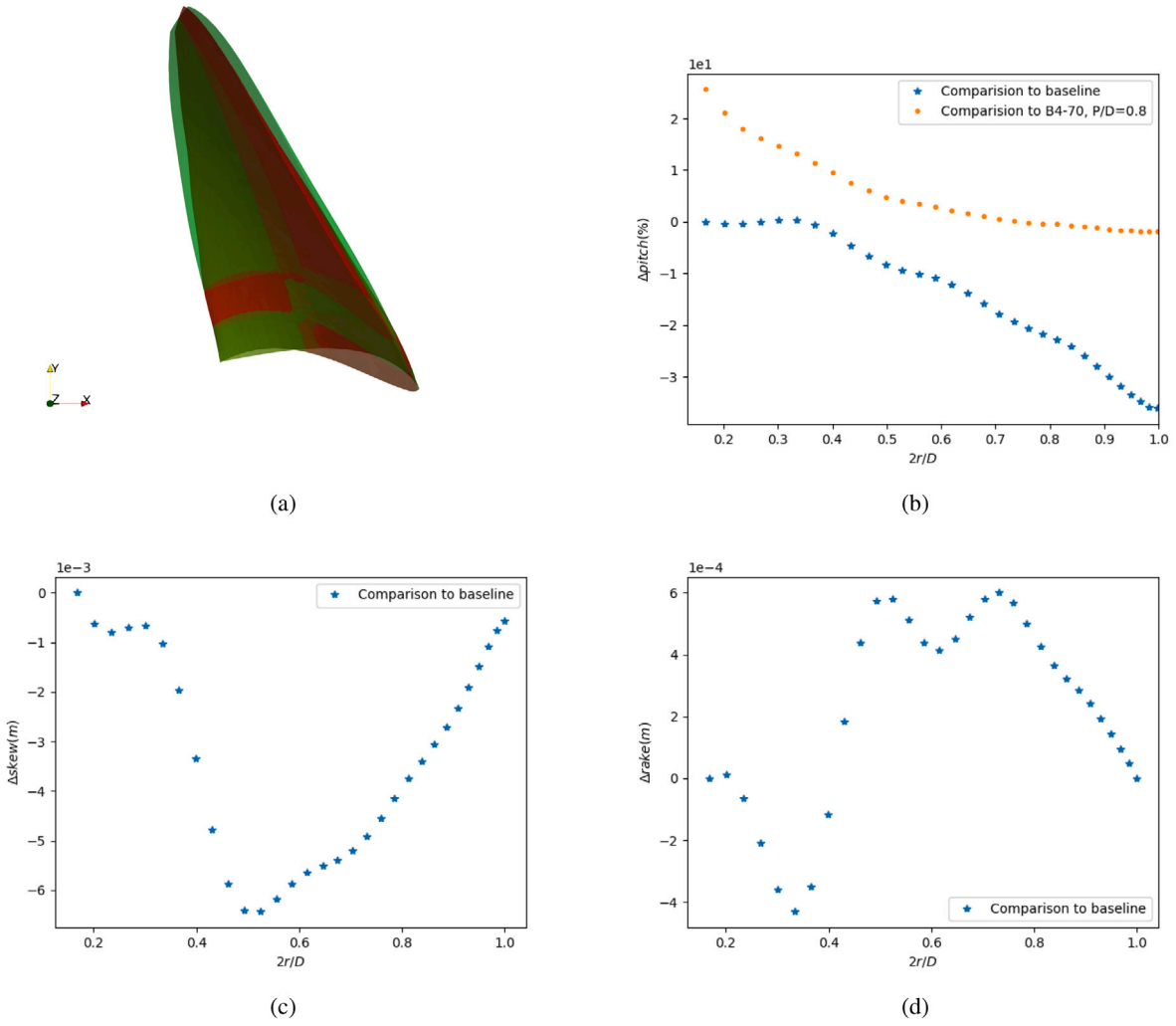

Fig. 10. Comparison of the optimum and baseline (a) geometry overlaid with the baseline Wageningen B4-70 P/D = 1.0 blade (green), (b) percentage change in pitch distributions of optimum, baseline and B4-70 P/D = 0.8 blade (c) delta skew distribution for baseline and optimum, (d) delta rake distribution for baseline and optimum.

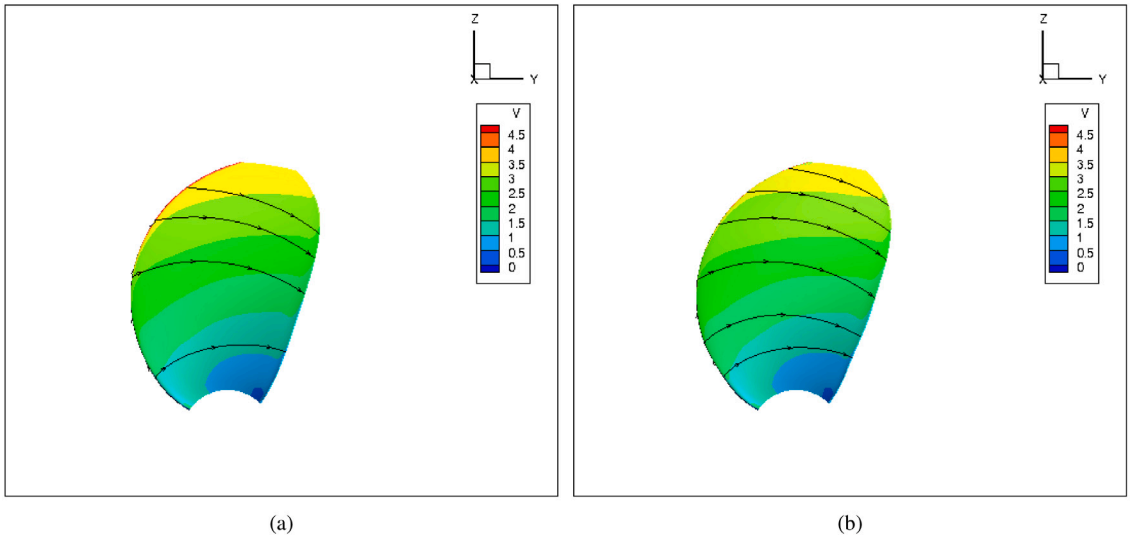

Fig. 11. Super-velocities (a) baseline and, (b) optimum.

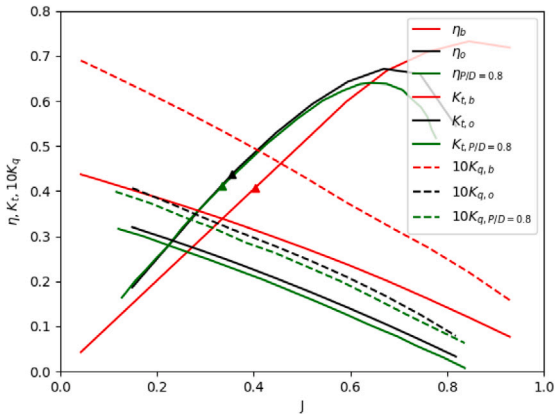

Fig. 12. Open water curves of optimum in Halton samples and baseline design. The markers are the point of operation for $v_{ship} = 1.72$ m s^{-1}, $T_o = 27$ kN.

Table 2
Optimum in Halton samples — parameters.

	Optimum (rad)
$\Delta\theta_{p,4}$	−0.0409
$\Delta\theta_{p,5}$	−0.0577
$\Delta\theta_{p,6}$	−0.0794
$\Delta\theta_{p,7}$	−0.0868
$\Delta\theta_{p,8}$	−0.0978
$\Delta\theta_{p,9}$	−0.1086
$\Delta\theta_{p,10}$	−0.1113

better than SVC with classical parameter only for $5 < k' < 11$. This could be because the RBF kernel for SVC tends to be a good feature map for classical design variables in the current demonstration case. Among the classifiers, NB has the highest precision score with $p = 0.9433$ for $k' = 8$ when using orthogonal parameters. For $6 \leq k' \leq 11$, all classifiers using orthogonal parameters outperform their counterparts that use classical design variables. For the domain $k' > 11$, it is inferred from silhouette scores that there is no additional geometric information, however, the dimensionality of the problem increases. This adversely affects classifier performance. Nonetheless, it is observed that NB with orthogonal parameters is the most reliable classifier for the identification of in-efficient designs. The reliability of NB with orthogonal parameters for multi-objective constrained optimization with supervised classifiers is demonstrated in a separate publication (Doijode et al., 2022)

The precision scores indicate that orthogonal parameters are better able to capture geometric variations and not obscure the relation between geometry, mesh and predicted performance. This addresses the first problem of multicollinearity. Furthermore, it enables more training and deploying more reliable machine learning models.

4.3. Dynamic optimization performance

To test the effectiveness of orthogonal parameters in identifying clusters of designs, it is run 50 times on the same Halton sample set. The goal of the test is to quantify possible gains of using orthogonal parameters versus classical design variables. The tests are run for two setups. In *Setup 1*, GMM in DO uses classical design variables for a-priori cluster identification and in *Setup 2*, it uses orthogonal parameters. In both setups, the supervised classifier is chosen to be the Naive-Bayes classifier due its high reliability. The following sections describe the optimum and the results of the test.

4.3.1. Optimum

The optimal solution in the Halton samples is shown in Fig. 10(a). The design parameters are listed in Table 2. Note that $\Delta\theta_{p,i} < 0.0$, this implies that the blade pitch has been reduced from the 4th–10th cross-section. $\Delta\theta$ is also the magnitude of depitching in radians. Thus, the optimum is found by depitching the blade.

The performance details are listed in Table 3. It is observed that the optimum delivers the target thrust of 27 kN for the specified $v_{ship} = 1.72$ m s^{-1} with higher efficiency than the baseline. The gains in efficiency are possible when axial, rotational and viscous losses are reduced (Schuiling and van Terwisga, 2016). Figs. 10(a)–10(d) compares the optimum to the baseline. The optimum's pitch distribution is also compared with a Wageningen B4-70 P/D = 0.8 blade as its efficiency at the operating point is better than the baseline. Furthermore, its open-water efficiency curve is closest to that of the optimum in the B4-70 series as shown in Fig. 12.

Fig. 10(b) shows that the pitch for the optimum is reduced at the mid-section and tip compared to the baseline. However, the pitch for the optimum at the root is higher than the Wageningen B4-70 P/D = 0.8 blade and similar at the tip. Due to the depitching, a reduction in rotational losses is expected. In order to generate the required thrust, the depitched propeller operates at a slightly higher rotation rate. As illustrated in Figs. 11(a)–11(b), despite the slightly higher rotation rate, the depitching results in lower super-velocities at the blade tip. This is particularly visible at the leading edge. The term super-velocity is used in aerospace design for regions where the flow is accelerated to higher velocities than incoming flow or moving geometries. Consequently, the skin friction is also lower reducing viscous losses. The resulting open-water curves and the operating point for the propellers are shown in Fig. 12.

Figs. 10(c)–10(d) show that skew and rake distribution for both propeller are very close. The differences are expected to result from the (a) B-Spline curve fit to the coordinates and, (b) surface lofting operation through the various fit curves. By having stricter convergence criteria for the B-Spline fit and lofting, it may be possible to reduce the difference even further.

Note, as indicated in Section 3, the results can quickly be verified and validated by using open-water curves of the Wageningen B4-70 propeller, which are readily available (Barnitsas et al., 1981). The baseline geometry corresponds to the Wageningen B4-70 propeller with P/D = 1.0. Thus, at the operating point propellers with a P/D < 1.0 can be found to be more efficient. Similarly, DO finds a better propeller by reducing the effective P/D of the baseline geometry. We recognize that the optimum in the sample set may not be the global optimum and that it is possible to choose a better starting P/D for the current operating point. Nonetheless, this is not expected to affect the comparison between orthogonal parameters and classical design variables.

4.3.2. DO Performance results

Fig. 13 illustrates the results for the design and optimization case with both classical and orthogonal design parameters. The x-axis shows the percentage of representative designs selected by the unsupervised classifier from each design cluster. The bars represent the ratio of the number of required iterations to the number of samples $N = 200$. The number at the head of each bar indicates the percentage of times the optimum design reported in Section 4.3.1 is found. The line indicates the mean performance increase of found solutions compared to the baseline design in percentage points.

It is observed that DO always finds designs that have an efficiency greater than the baseline Wageningen B4-70 propeller for both classical and orthogonal design variables within the Halton samples. DO performs better with orthogonal parameters. It tends to find the optimum design more often than its classical counterpart. Furthermore by evaluating under 30% ($\lambda = 20\%$) of the samples, solutions with an increase in efficiency of 3 percentage points are found. When ($\lambda > 30\%$), DO with orthogonal parameters finds the optimum in the Halton samples with a probability of 100%. On the other hand, DO with

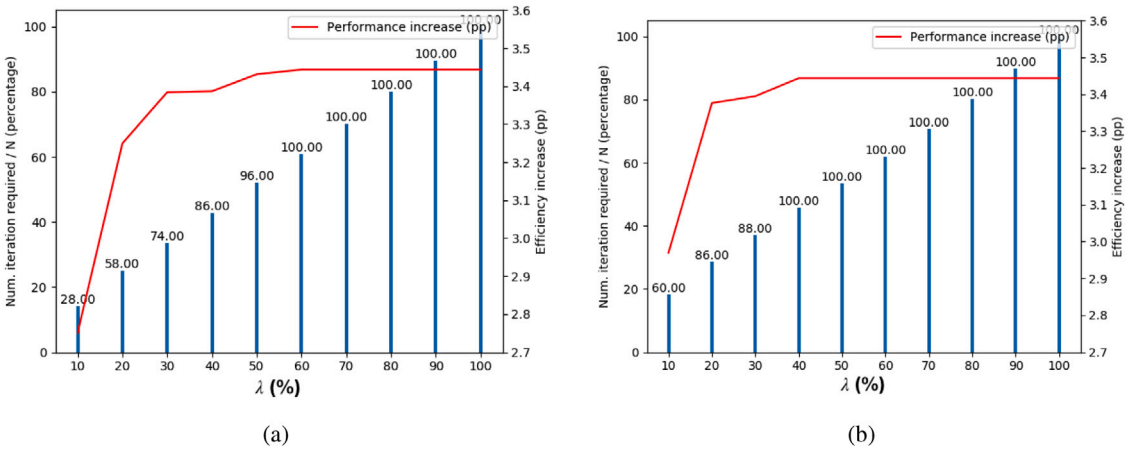

Fig. 13. Dynamic Optimization performance results for (a) Classical design variables — Setup 1 and (b) Orthogonal parameters — Setup 2. $\lambda = (s_i/s)/100.0$.

Table 3
Optimum in Halton samples — performance.

	Units	Wageningen B4-70, P/D = 1.0	Wageningen B4-70, P/D = 0.8	Optimum
J	–	0.4035	0.3353	0.3553
n	rps	1.9800	2.3910	2.2500
K_t	–	0.3133	0.2374	0.2439
K_q	–	0.0493	0.0308	0.0315
T (eq. constraint)	kN	27.0000	27.0000	27.0000
η (objective)	–	40.77%	41.18%	43.74%

classical counterparts needs more than 50% of representative designs to be selected in order to find the optimum with 100% probability.

Fig. 13 also contains information regarding convergence of DO. After λ is specified, the designs to evaluate from each cluster are determined at random. This makes DO for single objective optimization stochastic. Therefore, the optimization is repeated 50 times for each specified λ. From Fig. 13 it can be inferred that the number of iterations to convergence depends on λ. Low values of λ result in quicker convergence, however, the probability of missing the lucrative cluster increases. For sufficiently high values of λ the number of iterations to convergence increases and the probability of finding the lucrative cluster and design also increases.

The results indicate that DO with orthogonal parameters is able to identify design clusters and also able to guide search to lucrative clusters more efficiently than when using classical design variables. This is expected to be an outcome of orthogonal parameters addressing the problem of multicollinearity. Furthermore, the orthogonal parameters enable more reliable machine learning models to be trained and deployed.

5. Conclusion

A new design and optimization method – *Dynamic Optimization* – is proposed that synergizes supervised and unsupervised learning for the efficient design and optimization of propellers. Compared to classical design variables, Dynamic Optimization (DO) with orthogonal parameters is more efficient in identifying lucrative clusters. This gain in efficiency is attributed to the reliability of its two core components: (a) an orthogonal parametric model and (b) classifiers. The orthogonal parameters are demonstrated to address the challenge of multicollinearity thus preserving the correlation between parameters, mesh and performance. They also enable more reliable classifiers to be trained and deployed. Reliable and explainable classifiers are able to guide search algorithms to lucrative regions in the design space with fewer design evaluations thus reducing the computational cost of optimization. However, it is noted that improving efficiency of design and optimization is accompanied by a trade-off with completeness and optimality of search as demonstrated on a D&O problem of the benchmark Wageningen B4-70 propeller. Nonetheless, for the current thrust and ship speed constrained open-water efficiency objective, DO finds designs that have a higher efficiency than the Wageningen B4-70 propeller at a much lower cost than Halton sampling. Furthermore, the results can be quickly verified and validated with the open-water curves of Wageningen B4-70 propeller. As a next step, the method is to be demonstrated on cavitating propellers in a wake-field as a representative design problem in the second part of this series.

CRediT authorship contribution statement

Pranav Sumanth Doijode: Research (conceptualization, methodology), Software, Writing. **Stefan Hickel:** Guidance, Supervision and Review. **Tom van Terwisga:** Guidance, Supervision and Review. **Klaas Visser:** Guidance, Supervision and Review.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors extend their gratitude to Ir. Evert -Jan Foeth and Dr. Thomas Scholcz of MARIN and acknowledge the valuable exchange of thoughts and ideas. The authors are also grateful to CRS (Cooperative Research Ships) for supporting this research with PROCAL.

References

Abedjan, Z., Boujemaa, N., Campbell, S., Casla, P., Chatterjea, S., Consoli, S., Costa-Soria, C., Czech, P., Despenic, M., Garattini, C., et al., 2019. Data science in healthcare: benefits, challenges and opportunities. In: Data Science for Healthcare. Springer, pp. 3–38.

Allen, M.P., 1997. The Problem of Multicollinearity. Springer US, Boston, MA, pp. 176–180.

Arapakopoulos, A., Polichshuk, R., Segizbayev, Z., Ospanov, S., Ginnis, A., Kostas, K., 2019. Parametric models for marine propellers. Ocean Eng. 192, 106595.

Barnitsas, M.M., Ray, D., Kinley. Kt, P., 1981. Kq and Efficiency Curves for the Wageningen B-Series Propellers. Technical report, University of Michigan.

Bertetta, D., Brizzolara, S., Gaggero, S., Viviani, M., Savio, L., 2012. CPP propeller cavitation and noise optimization at different pitches with panel code and validation by cavitation tunnel measurements. Ocean Eng. 53, 177–195.

Bishop, C.M., 2006. Pattern Recognition and Machine Learning. Springer.

Bosschers, J., 2009. Investigation of hull pressure fluctuations generated by cavitating vortices. In: Proc. First Symposium on Marine Propulsors.

Brizzolara, S., Gaggero, S., 2009. Silent propellers with unconventional profile shapes. Examples obtained with a new automatic optimization method. In: Proceedings of the 4th Annual Conference MAST2009 on Maritime Systems and Technology. Stockholm, Sweden.

Chang, C.-C., Lin, C.-J., 2011. Libsvm: A library for support vector machines. ACM Trans. Intell. Syst. Technol. (TIST) 2 (3), 1–27.

Diez, M., Serani, A., Campana, E.F., Volpi, S., Stern, F., 2016. Design space dimensionality reduction for single-and multi-disciplinary shape optimization. In: 17th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference. p. 4295.

Doijode, P.S., Hickel, S., van Terwisga, T., Visser, K., 2022. A machine learning approach for propeller design and optimization: Part II. Appl. Ocean Res. *Accepted for publication.*

Esmailian, E., Ghassemi, H., Zakerdoost, H., 2017. Systematic probabilistic design methodology for simultaneously optimizing the ship hull–propeller system. Int. J. Naval Arch. Ocean Eng. 9 (3), 246–255.

Gaggero, S., Tani, G., Villa, D., Viviani, M., Ausonio, P., Travi, P., Bizzarri, G., Serra, F., 2017. Efficient and multi-objective cavitating propeller optimization: An application to a high-speed craft. Appl. Ocean Res. 64, 31–57.

Garg, N., Kenway, G.K., Martins, J.R., Young, Y.L., 2017. High-fidelity multipoint hydrostructural optimization of a 3-D hydrofoil. J. Fluids Struct. 71, 15–39.

Halton, J.H., 1960. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. Numer. Math. 2 (1), 84–90.

IMO, 2019. Reducing greenhouse gas emissions from ships. Accessed online on 2019-09-13. http://www.imo.org/en/mediacentre/hottopics/pages/reducing-greenhouse-gas-emissions-from-ships.aspx.

Knerr, S., Personnaz, L., Dreyfus, G., 1990. Single-layer learning revisited: a stepwise procedure for building and training a neural network. In: Neurocomputing. Springer, pp. 41–50.

LaValle, S.M., 2006. Planning Algorithms. Cambridge University Press.

Liu, Z., Young, Y.L., 2009. Utilization of bend–twist coupling for performance enhancement of composite marine propellers. J. Fluids Struct. 25 (6), 1102–1116.

Park, G.-J., Lee, T.-H., Lee, K.H., Hwang, K.-H., 2006. Robust design: An overview. AIAA J. 44 (1), 181–191.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine learning in python. J. Mach. Learn. Res. 12, 2825–2830.

Peri, D., 2012. Conformal free form deformation for the optimisation of complex geometries. Ship Technol. Res. 59 (1), 36–41.

Prautzsch, H., Boehm, W., Paluszny, M., 2002. Bézier and B-Spline Techniques, Volume 6. Springer.

Rasmussen, C.E., Williams, C.K., 2006. Gaussian Processes for Machine Learning. Springer.

Rousseeuw, P.J., 1987. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. J. Comput. Appl. Math. 20, 53–65.

Sarkar, S., Dong, A., Gero, J.S., 2009. Design optimization problem reformulation using singular value decomposition. J. Mech. Des. 131 (8), 081006.

Schuiling, B., van Terwisga, T., 2016. Energy analysis of a propeller in open water using a RANS method. In: 24th International HISWA Symposium on Yacht Design and Yacht Construction. Amsterdam, The Netherlands.

van Terwisga, T., van Wijngaarden, E., Bosschers, J., Kuiper, G., 2007. Achievements and challenges in cavitation research on ship propellers. Int. Shipbuild. Prog. 54 (2–3), 165–187.

Theotokatos, G., Tzelepis, V., 2015. A computational study on the performance and emission parameters mapping of a ship propulsion system. Proc. Inst. Mech. Eng. Part M 229 (1), 58–76.

Van Loan, C.F., Golub, G., 1996. Matrix computations (johns hopkins studies in mathematical sciences).

Vaz, G., Bosschers, J., 2006. Modelling three dimensional sheet cavitation on marine propellers using a boundary element method. In: Sixth International Symposium on Cavitation CAV2006. Wageningen, The Netherlands.

Vaz, G., Hally, D., Huuva, T., Bulten, N., Muller, P., Becchi, P., Herrer, J.L., Whitworth, S., Macé, R., Korsström, A., 2015. Cavitating flow calculations for the e779a propeller in open water and behind conditions: code comparison and solution validation. In: Proceedings of the 4th International Symposium on Marine Propulsors (SMP'15), Austin, TX, USA, pp. 330–345.

Vesting, F., 2015. Marine Propeller Optimisation-Strategy and Algorithm Development (Ph.D. thesis). Chalmers University of Technology.

Vesting, F., Johansson, R., Bensow, R., 2013. Parameter influence analysis in propeller optimisation. In: Binns, J., Brown, R., Bose, N. (Eds.), Proceedings of the Third International Symposium on Marine Propulsors. University of Tasmania, Launceston, Tasmania, Australia, pp. 397–404.